UNIVERSITY OF KWAZULU-NATAL

College of Agriculture, Engineering and Science

Human Action Recognition Using Spatial-Temporal Analysis

By:

Denver Naidoo

211520771

Supervisor

Prof Tom Walingo

Co - Supervisor:

Prof Jules-Raymond Tapamo

In fulfilment of the academic requirements of the degree of Master of Science in Engineering (Electronic Engineering), School of Engineering, University of KwaZulu-Natal

Examiners Copy

November 2019

PREFACE

The research contained in this dissertation was completed by Denver Naidoo, at the Discipline of Electrical, Electronic and Computer Engineering, School of Engineering, College of Agriculture, Engineering and Science, University of KwaZulu-Natal, Howard College, South Africa under the supervision of Prof Jules-Raymond Tapamo and Prof Tom Walingo.

I hereby declare that the contents of this work have not been submitted in any form to another university and, except where the work of others is acknowledged in the text, the results reported are due to my investigations.

____

Denver Naidoo

Declaration – Supervisor

As the candidate's supervisor I agree to the submission of this dissertation.

____

Prof Tom Walingo

Declaration – Co-Supervisor

As the candidate's co-supervisor I agree to the submission of this dissertation.

____

Prof Jules-Raymond Tapamo

Declaration – Plagiarism

I, Denver Naidoo, declare that,

1. The research reported in this dissertation, except where otherwise indicated, is my original research.
2. This dissertation has not been submitted for any degree or examination at any other university.
3. This dissertation does not contain other persons' data, pictures, graphs or other information, unless specifically acknowledged as being sourced from other persons.
4. This dissertation does not contain another persons' writing, unless specifically acknowledged as being sourced from other researchers. Where other written sources have been quoted, then:
   a) Their words have been re-written but the general information attributed to them has been referenced.
   b) Where their exact words have been used, then their writing has been placed in italics and inside quotation marks, and referenced.
5. This dissertation does not contain text, graphics or tables copied and pasted from the Internet, unless specifically acknowledged, and the source being detailed in the dissertation and in the reference's sections.

Denver Naidoo

Declaration – Publications

Details of contribution to publications that form part of, and/or include research presented in this dissertation:

1. Denver Naidoo, Jules Raymond Tapamo and Tom Walingo, "Human Action Recognition using Spatial-Temporal Analysis and Bag of Visual Words", In Proceedings of t*he 14th International Conference on Signal Image Technology and Internet Based Systems (SITIS)*, ISBN 978-1-5386-9385-8, Las Palmas de Gran Canaria, Spain, 26-29 November 2018, pp 697–702.

2. Denver Naidoo, Jules-Raymond Tapamo, and Tom Walingo, Combining Motion History Images and Deep Learning for Human Activity Recognition (Manuscript in preparation)

____

Denver Naidoo

Acknowledgement

I would like to thank my supervisors, Prof Tom Walingo and Prof Jules-Raymond Tapamo for their guidance and support during the course of this research. I would also like to thank my wife and parents for their support during this research.

ABSTRACT

In the past few decades' human action recognition (HAR) from video has gained a lot of attention in the computer vision domain. The analysis of human activities in videos span a variety of applications including security and surveillance, entertainment, and the monitoring of the elderly. The task of recognizing human actions in any scenario is a difficult and complex one which is characterized by challenges such as self-occlusion, noisy backgrounds and variations in illumination. However, literature provides various techniques and approaches for action recognition which deal with these challenges. This dissertation focuses on a holistic approach to the human action recognition problem with specific emphasis on spatial-temporal analysis.

Spatial-temporal analysis is achieved by using the Motion History Image (MHI) approach to solve the human action recognition problem. Three variants of MHI are investigated, these are: Original MHI, Modified MHI and Timed MHI. An MHI is a single image describing a silhouettes motion over a period of time. Brighter pixels in the resultant MHI show the most recent movement/motion. One of the key problems of MHI is that it is not easy to know the conditions needed to obtain an MHI silhouette that will result in a high recognition rate for action recognition. These conditions are often neglected and thus pose a problem for human action recognition systems as they could affect their overall performance.

Two methods are proposed to solve the human action recognition problem and to show the conditions needed to obtain high recognition rates using the MHI approach. The first uses the concept of MHI with the Bag of Visual Words (BOVW) approach to recognize human actions. The second approach combines MHI with Local Binary Patterns (LBP). The Weizmann and KTH datasets are then used to validate the proposed methods.

Results from experiments show promising recognition rates when compared to some existing methods. The BOVW approach used in combination with the three variants of MHI achieved the highest recognition rates compared to the LBP method. The original MHI method resulted in the highest recognition rate of 87% on the Weizmann dataset and an 81.6% recognition rate is achieved on the KTH dataset using the Modified MHI approach.

CONTENTS

ABBREVIATIONS

BOVW: Bag of Visual Words

CNN: Convolutional Neural Network

DoG: Difference of Gaussian

DTW: Dynamic Time Warping

FLDA: Fischer Linear Discriminant Analysis

HAR: Human Action Recognition

HMM: Hidden Markov Models

HOG: Histogram of Oriented Gradients

KNN: K-Nearest Neighbours

LBP: Local Binary Patterns

LBP-TOP: Local Binary Patterns in Three Orthogonal Planes

LDA: Linear Discriminant Analysis

LoG: Laplacian of Gaussian

MEI: Motion Energy Images

MHI: Motion History Images

PCA: Principle Component Analysis

SIFT: Scale Invariant Feature Transform

SURF: Speed up Robust Features

SVM: Support Vector Machines

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1 - INTRODUCTION

Every day, we perform actions and activities to accomplish some purpose, regardless of how insignificant those actions/activities may be. For instance, a person playing a sport, such as cricket responds and interacts with the environment using his/her legs, arms, torso etc. The human vision system can easily identify these actions and activities. However, in the computer vision domain, this can be a very difficult task to accomplish. In the past two decades, a lot of attention has been given to the human action recognition problem, so much so that it has become one of the most researched areas in computer vision, machine learning and pattern recognition in recent times. This is due to its many potential applications, such as monitoring of patients or the elderly, entertainment, automated surveillance systems, human computer interaction and health care systems [1] [2] [3] [4]. Some of the reasons why human action recognition has gained a lot of attention include events such as the September 11th aeroplane hijackings and the Boston marathon bombings. The rapid spread of surveillance cameras has deemed the manual analysis of video data to be a time-consuming and an expensive task. By using computer vision and artificial intelligence, this task can be made cost effective and the analysis of large video data easier.

In this chapter, a brief introduction into the history of motion is presented in section 1.1. Section 1.2 provides the motivation behind this research. In section 1.3, the problem statement is defined. Section 1.4 presents the main goals and specific objects for this work. In section 1.5 the contributions made in this work is established. Section 1.6 provides the dissertation outline for this work.

1.1 Brief History of Human Motion

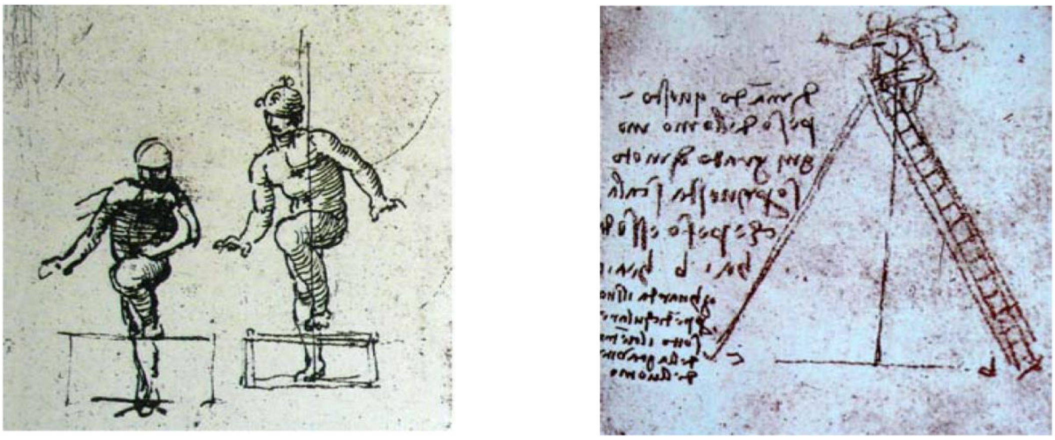
The interest in human motion goes far back in human history. Many disciplines, such as, Mathematics (Functional analysis, Probability, Statistics, Linear algebra, Geometry, etc.), Science (Physics, Chemistry, Botany, Ecology, Biology, etc.), Medicine (Cardiology, anatomy, etc.) and art (sculptures, paintings etc.) have been, and continue to be interested in different aspects of motion [5].

1.1.1 Artistic Representation

One of the first artists to take interest in human representation was Leonardo Da Vinci. In one of his sketch books (referring to Figures 1.1a and 1.1b), Da Vinci (1452–1519) wrote [5]:

"it is indispensable for a painter, to become totally familiar with the anatomy of nerves, bones, muscles, and sinews, such that he understands for their various motions and stresses, which sinews or which muscle causes a particular motion"

"*I ask for the weight [pressure] of this man for every segment of motion when climbing those stairs, and for the weight he places on b and on c. Note the vertical line below the centre of mass of this man."*



(a) **(b)**

Figure 1.1 - (a) Man Going up a Stairs, (b) Man climbing up a ladder [5]

1.1.2 Biomechanics

Giovanni Alfonso Borelli (1608–1679) was the first to understand that muscles function according to mathematical principles and bones function as levers [5]. His studies included muscle analysis and mathematical representations of movements such as running or jumping as shown in Figure 1.2.

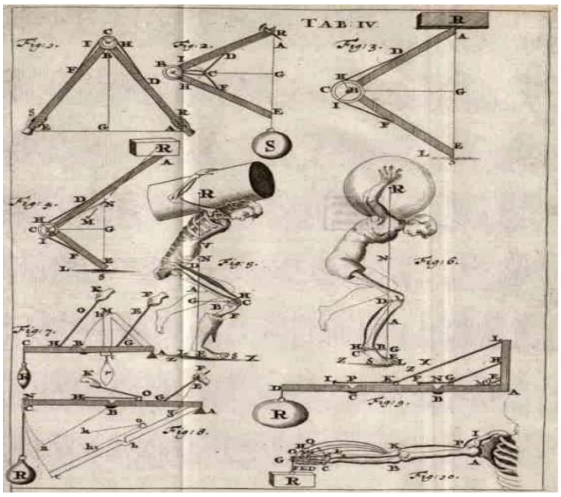


Figure 1.2 - Copper engraving by Borelli in 1680/1681 (Deutsches Museum, Munich) [5].

1.1.3 Motion Perception

Etienne-Jules Marey (1830–1904) made experiments that were influential in the field of cinematography [5] [6]. His research was based on the process of capturing and displaying moving images. He conducted motion studies through which he collected data using various instruments, which can be seen in Figure 1.3. He also made movies that were at a high speed of 60 images per second and of excellent quality.

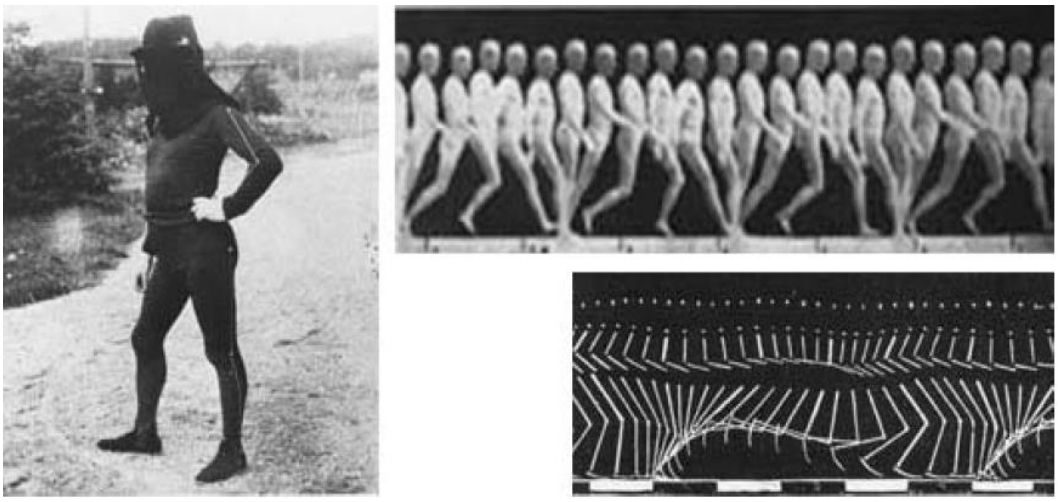


Figure 1.3 - Left: a man in a black dress; limbs are marked by white lines. Right, top: a chronophotograph of a striding man dressed partially in white, and partially in black. Right, bottom: white lines in a chronophotograph of a runner [5] [6]

Gunnar Johansson [7] [8] was at the forefront of studies for programmed human motion analysis based on the use of image sequences. He used LED's ("Moving Light displays) to identify people. Figure 1.4 shows an instance of the use of moving light display of a man walking.

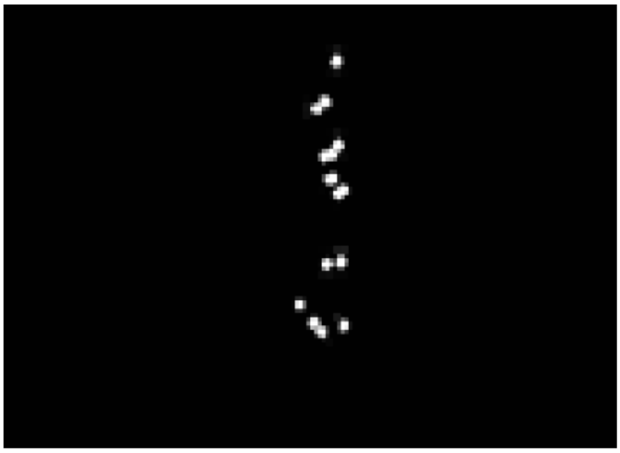


Figure 1.4 - Gunnar Johansson use of moving light display for motion of a man walking [7]

1.2 Motivation

The need for systems to recognize human actions has become more apparent in the world we are living in. There are various applications in security, health, entertainment and the robotic industries requiring action recognition systems to reduce costs and to aid when there is not enough manpower to monitor video captured in order to take adequate action, or to use intelligent robots to penetrate spaces were human life could be threatened.

1.2.1 Security/Surveillance

Today, surveillance cameras can be found at almost every corner of a city, and their major purpose is to increase the security level by detecting unwanted behaviours, such as violence, theft and vandalism. However, due to the large amount of surveillance data captured from these cameras, it becomes tedious and almost impossible for human operators to analyse the footage or to assess them in real time in an error free manner. The camera feed is often monitored by an operator in real time or through recorded footage in order to detect abnormal behaviour. Whilst this method has worked to an extent, it remains tedious and prone to human errors. Automated human action recognition can play a vital role in aiding security personnel to identify a person's criminal actions and alert the relevant authorities swiftly. By using action recognition systems, surveillance cameras can be analysed 24 hours a day, 7 days a week, without needing much human intervention. The use of automated human action recognition for monitoring unwanted criminal behaviour can help alert security personal instantly when a crime has taken place. Human action recognition in security and surveillance systems can greatly improve safety and help reduce crime rates by helping to identify dangerous situations.

1.2.2 Health care

Action recognition in health care can aid doctors in monitoring a patient's state of well-being by detecting potential physical and mental problems early. For instance, hospitals that are under staffed can use action recognition systems to monitor their patients' physical state (their eating habits, walking and sleeping patterns, etc.) and alert medical doctors should a problem arise. A practical example is the case of patients who suffer from diseases such as Alzheimer's and dementia, who are known to be more susceptible to falling. An action recognition system can be used to detect this incident and alert the relevant hospital staff immediately.

1.2.3 Robotics

Robotics combines branches of science and engineering and is found in various industries such as manufacturing, health care and mining. Action recognition can be very useful in robotics. For example, a domestic service robot can clean a floor after seeing that a person has performed the action of "cooking in a kitchen" [9].

1.2.4 Autonomous vehicles

Human action prediction approaches have the possibility of becoming very important in building autonomous vehicles. Action prediction approaches can determine a person's intention in a short space of time. The use of action prediction in autonomous vehicles can enable the vehicle to predict a pedestrian's future motion and path, which could be critical in avoiding an accident [10].

1.2.5 Human Computer Interaction

Human computer interaction deals with how users/people interface with computer technology. Today, people interact with computer devices in various ways, such as voice recognition, eye tracking (e.g. instead of using a mouse to control a pc), virtual reality, etc. Action recognition can be used in this field to recognize human actions aimed at performing certain tasks on a computer, game or, simulator [11]. Training flight deck controllers to conduct helicopter training can be an expensive task. Human action recognition can be used to analyse the deck controllers' motion and feed the actions that were performed into a helicopter training simulator to allow for dual training of a pilot and a deck controller [11]. This will save a lot money and as well as the effort required to train deck landing controllers and pilots, improving the overall competency of the above-mentioned personnel. The Microsoft Kinect sensor [12] [13] is one of the most popular gesture recognition tools found in millions of homes. This device senses motion and allows users to interact and control user interfaces on a television/computer using gestures. The sensor however is limited to recognizing short actions or gestures. Action recognition using video can allow people to use their phone cameras or cheap cameras at home to interface with devices such as televisions and computers instead of purchasing additional equipment such as the Kinect.

1.2.6 Entertainment

Gaming attracts a large number of people due to its entertainment value and as computing and graphics devices have become more and more powerful in the recent years, the gaming industry has grown vastly [10]. Newer games whose gameplay involves the entire body, such as sport and dancing games have caught the interest of people of various ages. In order to achieve accurate recognition of human actions, these games rely on RGB-D sensors such as the Kinect from Microsoft and the Wii's handheld sensors used for tracking a person's movement.

Recognizing human actions is a difficult task, and as a result, the above-mentioned applications require uniquely modified systems for recognizing specific movements in a given domain. For example, a person swinging a cricket bat to cause harm to someone may be very difficult for a system to distinguish from a normal cricket shot or a malicious intent to cause injury. Apart from each of the above-mentioned applications having its own problems for action recognition, the computer vision domain presents various challenges for recognizing a movement, and even more so, an action or activity.

1.3 Problem Statement

In dealing with human action recognition (HAR), many challenges arise, including actions/activities that have to be recognised against noisy images, noise in backgrounds and losses in feature extraction. Even though there is no complete system for human action recognition, a lot of effort has been made to solve the individual action recognition problems. To state the problem in simpler terms, given a sequence of images/video with one or more persons performing an action, can a system be designed to automatically recognize what action is being performed? In order to better understand the human action recognition problem, the terms "Action" and "Activity" need to be defined. Although the terms action and activity are often interchanged, they have different meanings as is given below:

- An action can be defined as a single movement by a person. For example, the waving of hands or jumping. Actions usually last a few seconds.
- Activities are more complex and can be defined as a sequence of actions being performed to accomplish a certain task. For example, playing sport or cooking.

One of the more recognized methods of representing human actions from a video is the MHI method [14]. The problem with MHI is that its relevant parameters are not so easy to tune. This plays a crucial role on how much of history information should be stored for feature extraction and results in an adverse effect on recognition rates. These parameters are often neglected and optimizations are done at other stages of a recognition system. This dissertation will take a closer look at MHI and its various conditions that will aid in a systems performance for action recognition and to propose a new method for solving the action recognition problem by combining MHI with Bag of Visual Words.

1.4 Main Goal and Specific Objectives

The main goals of this dissertation are to investigate the conditions under which human action recognition can be efficiently performed using MHI and to propose a new method for human action recognition by using the Bag of Visual Words approach.

The specific objectives of this work are as follows:

- Conduct a literature review of current human action recognition techniques.
- Propose a method for solving the action recognition problem using Local Binary Patterns and Bag of Visual Words.
- Investigate the effects of the relevant parameters of MHI in order to improve recognition rates of human action recognition. These parameters determine how much motion information to keep for an MHI, and play a crucial role in the human action recognition system's overall recognition rate.
- Compare the results achieved to the state-of-the-art

1.5 Contributions

This work contributes to the existing domain knowledge in the following ways:

- Propose a new HAR method that combines MHI and Bag of Visual Words to obtain feature vectors for training.
- Demonstrate the effects of using Local Binary Patterns (LBP) with MHI for human action recognition.
- Revisit MHI for human action recognition with respect to the selection of parameters that achieve better recognition rates.

1.6 Dissertation outline

In Chapter 2, a full literature review is presented. The review presents current methods of spatial-temporal analysis and a general review of various techniques used for human action recognition.

Chapter 3 introduces the concept of Motion History Images and its mechanics. Variants of the Motion History Image algorithm are presented and their strengths and shortcomings are outlined. This chapter forms part of the basic foundation for the proposed method of using MHI with BOVW.

Chapter 4 introduces the proposed approach namely MHI using BOVW. A common approach for human action recognition is introduced which is MHI using LBP. Various feature extraction methods that can be used to obtain feature vectors from the resultant MHI are also investigated in this chapter. The feature extraction methods are: Local Binary Patterns (LBP), Speed Up Robust Features (SURF), Scale Invariant Feature Transform (SIFT) and Bag of Visual Words (BOVW).

Chapter 5 presents the experiments and discusses the results achieved from the two approaches: MHI using BOVW and MHI using LBP. The experiments also show the overall recognition rates achieved for these two methods when the relevant parameters of MHI is varied.

Chapter 6 concludes the dissertation and outlines future works.

CHAPTER 2 - LITERATURE REVIEW

2.1 Introduction

Recognizing human actions is a fundamental problem in the computer vision and artificial intelligence domains. It lends itself to many practical applications [1] [2] [15]. The aim of human action recognition is to recognize human actions from a video stream or a sequence of images. The recognition of the identity of a person's activity, based on his/her global body structure and the global body dynamics are discussed in existing sources. Of particular interest regarding identity recognition of an activity is the human gait [16]. Other approaches using global body structure and dynamics are concerned with the recognition of simple actions such as running and walking. Almost all methods are silhouette or contour-based. Subsequent techniques are mostly holistic (the entire silhouette or contour is evaluated without taking individual body parts into account). Dimensionality reduction and spatiotemporal analysis examines the entire human silhouette or image. It does not consider each part of the human and is therefore considered as a holistic approach. Many approaches for human activity recognition have been proposed in the literature. Good surveys on the subject can be found in [2] [4] [17]. Given the promising endeavour's taken to develop fool proof systems for human action recognition, it still faces many challenges. These include:

- **Each action has varied dimensionality and data redundancy**:
  When an action video is subdivided into spatio-temporal patches, it may cause high dimensional data samples. Data redundancy may also occur from the high temporal sampling rate [3].
- **Some actions/activities share common movements:**
  Actions such as jogging, running and walking share similar movements and may be very difficult to distinguish. This can be seen when using the famous Weizmann and KTH datasets. It is noticeable from these datasets that some running actions by some subjects/people look like jogging and vice versa [3].
- **Variations in illumination:**
  With famous datasets such as KTH and Weizmann, lighting conditions are usually fixed. Therefore, indoor and constant illuminations produce easier datasets to evaluate. This is not true in reality because of variations in light during the day, for example,

sunlight in the morning vs sunlight at noon. Weather variations also cause different lighting conditions, for example, cloudy vs raining vs sunny. Due to these large variations in lighting, the recognition process, thus becomes more difficult [3].

- **Occlusion Issue:**
  The presence of cluttered backgrounds in a video scene makes action analysis a difficult task [3].

In this chapter, a closer look in to some common techniques for human action recognition will be reviewed in detail. Section 2.2 reviews human action recognition using Dimensionality Reduction. Section 2.3 introduces human action recognition using Spatio-Temporal Analysis. In Section 2.4 a review of human action recognition using Hidden Markov Models is conducted. Section 2.5 explores human action recognition using Deep Learning. Section 2.6 reviews human action recognition using other methods that are commonly used. Section 2.7 summarises the results found in literature.

2.2 Human Action Recognition using Dimensionality Reduction

High data dimensionality leads to significant issues with regards to the robustness and accuracy of recognition due to insufficient knowledge about the data population and a limited number of training samples. Dimensionality reduction thus becomes a separate and possibly the most critical task of a recognition system. Dimensionality reduction functioning as a feature extraction technique has two objectives. The first objective is to reduce the computational complexity of the subsequent classification with minimal loss of classification accuracy. The second objective is to circumvent the generalization problem of the subsequent classification and hence, enhance its accuracy and robustness. Many works have used various linear subspace methods to achieve dimensionality reduction [15] [18] [19]. Methods such as Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA) and Fisher Linear Discriminant Analysis (FLDA) form an integral part of a recognition system as they first reduce the dimension and then, extract crucial features needed for the system.

The problem with image representation is its high dimensionality. Consider a two-dimensional p x q grayscale image that spans in a $m = pq$-dimensional vector space, an image with 100 x 100 pixels will reside in a 10000 - dimensional image/vector space. The question is: are all dimensions equally useful for us? A decision can only be made if there is some variance in the data. Using dimensionality reduction techniques, the components that account for the most information can be kept [20].

2.2.1 Eigenfeature Regularization and Extraction

The aim of eigenfeature regularization and extraction is to replace the original eigenvalues with modelled eigenvalues which make regularizing eigenfeatures corresponding to small and zero eigenvalues easier. The model is derived from the within-class (action set) scatter matrix. It also aims to fulfil a discriminant assessment in the entire eigenspace. This is accomplished from the intensity of the data found in the action video stream [15].

Bappaditya et al. [15] [21] proposed a holistic based eigenfeature regularization approach based on a 3-parameter model. Since the human body is not a rigid object and can present many shapes and postures along with self-occlusions, a robust modelling of the object is difficult to obtain. The major challenges to such models are viewpoint dependence, appearance variability, presence of occlusion, statistical or deterministic representation of a sequence of motion segments, and a parsing mechanism that can temporally align the input signal with known activity patterns [15].

The result of the discriminant evaluation according to the Fisher criteria is not directly applicable to the human action recognition area because the scatter matrix is often singular, this is due to the limited number of training samples and temporal dependencies in the image capture of the same activity [15].

Eigenfeature regularization and extraction aim to address the aforementioned problems. In the 3-parameter case the Eigen spectrum is modelled such that the principal and noise space have a slower decay compared to the real Eigen spectrum.

The experiments conducted for this approach showed very high accuracy in recognition rates. Tests were done on different datasets and the 3-parameter eigenfeature regularization and extraction method out-performed PCA, FLDA and the 2-parameter Eigenfeature Feature Regularization Extraction methods. The method however under-performs for certain activities when using the Dynamic Time Warping (DTW) process.

Lee et al. [18] used the 2-parameter model for Eigen Feature Regularization and Extraction proposed by Jiang et al. [21] to perform action recognition on gait sequences. The authors used motion contour images as inputs into this dimensionality reduction feature extraction technique. The method is restricted to the recognition of the action associated with walking and hence, its ability to recognize other actions is unknown.

2.2.2 Principal Component Analysis

Principal component analysis (PCA) is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components. The number of principal components is less than or equal to the number of original variables or the number of observations. This approach is designed where the first principle component computed has the largest possible variance. It accounts for as much variability in the data as possible. Each succeeding component computed after that in turn has the highest variance possible under the constraint that it is orthogonal to the preceding components. An uncorrelated orthogonal basis set is then formed in to vectors. PCA is sensitive to the relative scaling of the original variables [22]. PCA works well in the image processing domain. Its purpose is to reduce the large dimensionality of the data space (observed variables) to the smaller intrinsic dimensionality of feature space (independent variables to describe the data economically. Using principle component analysis, feature extraction and training can use less data, allowing for faster computation and lower memory usage.

Jyotsna et al. [23] used a silhouette-based approach together with PCA and Independent Component Analysis (ICA) to recognize human actions. By using PCA, the computation is less complex and the feature extraction step can be done much faster. However, by using more optimized subspace methods, the recognition performance of the human action recognition system can be further improved. Their method is limited to 4 action classes being performed from the Weizmann dataset. A more appropriate test would be to test the entire dataset which includes 10 action classes. This would test the robustness of their proposed method and how well it would be able to distinguish between run, jump and walk actions, which share similar movement traits.

2.2.3 Action recognition using other subspace methods

Shoa and Chen [24] used spectral discriminant analysis on silhouette based human action recognition. The Bag of Words model was combined with the Histogram of body poses sampled from silhouettes in the video sequence to recognize human actions. This method of discriminant analysis achieved a high recognition rate. However, it must be noted that testing was done using the Weizmann dataset only. To boost the robustness of the method, testing should be done on other datasets.

Wang and Suter [25] exploited Local Preserving Patterns (LPP) for dimensionality reduction. This led to low-dimensional embedding of human actions. By using LPP, silhouettes of an associated video were projected into a low-dimensional space and therefore the spatiotemporal property of the action can be characterized. This also preserved the geometric structure of the action. The median Hausdorff distance or normalized spatiotemporal correlation was used for similarity measures to match the embedded actions. K-Nearest Neighbours (KNN) was then used to classify the actions. The proposed method however did not take into account both kinematic and shape information. A fusion of both approaches would possibly improve the accuracy and reliability of the system.

2.3 Human Action Recognition using Spatio-Temporal Analysis

2.3.1 Motion History Images and Motion Energy Images

The first steps to spatiotemporal analysis were taken by Bobick and Davis [14]. Motion Energy Images (MEI) and Motion History Images (MHI) were used as temporal templates to recognize aerobic movements. The basis of the representation was a temporal template or a static vector image where the vector value at each point is a function of the motion properties at the corresponding spatial location in an image sequence. Matching was done using the seven Hu invariant moments approach. A challenge to this approach is the condition when one person partially occludes another, making separation difficult, if not impossible [14].

Bradski and Davis [26] modified the basic MHI [14] to encode the actual time directly using a floating-point representation in the format (seconds, milliseconds). This representation proved to be independent of the system speed and frame rates. This representation can be used to determine the current pose of the object and to segment and measure the motions induced by the object in a video scene. The segmented regions are not “motion blobs”, but rather motion regions naturally connected to the moving parts of the object of interest. Bradski and Davis [26] then used Hu moments to recognize pose. The gradient of the Timed Motion History Image (TMHI) was used to determine the normal optical flow (e.g. motion flow orthogonal to object boundaries). The motion was then segmented relative to object boundaries and the motion orientation and magnitude of each region are obtained [26]. Although their method achieved a high recognition rate, it was not tested on more challenging datasets, such as Weizmann, KTH and the INRIA IXMAS datasets.

Davis [27] later proposed a hierarchical extension to motion history images in order to compute dense motion flow from the extracted MHI. Hierarchical MHI was then achieved by constructing image pyramids from the gradients. The pyramids were obtained by recursively computing a low pass filter and sub-sampling the image. Motion gradients were calculated to obtain the speeds of the motion from the motion template. The motion gradients obtained were then convolved at each level of the pyramid. Davis [27] then used polar histograms to categorize the action. The method aimed to address the sensitivities of the occlusion issue and the limitations that arose from global analysis. Although the issue of occlusion was addressed, one limitation of this method was that it depends on the background being static, and thus may need to be tested on datasets that have dynamic backgrounds.

A 3D extension of the temporal templates was proposed by Weinland et al. [28]. The authors used multiple cameras to build motion history volumes (MHV) and performed action classification using Fourier analysis in cylindrical coordinates. An obvious problem for the false detections, is the nearly infinite class of possible motions. Modelling unknown motions may require more than a single threshold and class. Multiple classes and explicit learning on samples of unknown motions thus becomes vitally important. Another problem was that many motions cannot be modelled by a single template. Small motions may seem very similar, but over time belong to very different actions. For example, the turnaround motion is split into several small steps that may easily be confused with a single side step. In such cases temporal networks over templates for example could be used to resolve these ambiguities. [28]

Kellokumpu et al. [1] proposed two methods for spatio-temporal analysis. The first one used MHI to capture movement dynamics and then used texture features to characterize the observed movements. They then extended this idea into a spatio-temporal space and described human movements with dynamic texture features. Local binary patterns and Three Orthogonal Planes of local binary patterns (LBP-TOP) were used as texture descriptors for feature extraction. Both methods were trained using Hidden Markov Models (HMM) and Support Vector Machines (SVM). The approaches were then evaluated using the KTH and Weizmann datasets.

Ahsan et al. [29] used Local Binary Patterns (LBP) to extract highlighted features from spatio-temporal templates, the features were computed as a histogram to construct feature vectors. Rather than using MHI, they use Directional MHI (DMHI) to formulate the motion template. Shape features were taken from selective silhouettes which were then concatenated with LBP histograms. The performance of the proposed action representation method along with some variants were evaluated using the Weizmann action dataset. A problem with their proposed method was that they did not incorporate any mechanism for scale and rotation invariance or view point changes. The proposed method also employed a basic 3x3 local binary pattern neighbourhood, which may not be the most optimal local binary pattern algorithm to use.

Naiel et al. [30] used MHI and MEI with 2-dimensional principal component analysis (2DPCA) to recognize actions from the Weizmann and INRIA IXMAS datasets. The 2-Dimensional Principal Component Analysis is more efficient in terms of computation and complexity compared to the conventional Principal Component Analysis. This approach however only considered the challenge of view invariant human action recognition and not challenges such as occlusions.

Meng et al. [31] propose a new approach to motion templates called the Hierarchical Motion History Histogram (HMHH). They combine the MHI with the new Hierarchical Motion History Histogram approach and extract a low dimension feature vector to be used in training Support Vector Machines.

Chuanzhen Li et al. [32] proposed a method for action recognition that takes advantage of both global and local representation of video sequences. Dense Harris corners were first extracted as local interest points and then masked using MHI. For global descriptors, they proposed a new method to estimate the distribution of rectangular patches which the authors call histogram of local rectangles (HLOR). In order to get enough temporal information, the Histogram of

Gradients (HOG) method was used on the extracted MHI. The feature vector obtained from both the HLOR and MHI-HOG methods were concatenated and trained using support vector machines. They achieve a recognition rate of 97.53% and 93.52% on the famous Weizmann and KTH datasets respectively. However, for this approach the descriptor would need to be extended to actions with complex backgrounds.

Huang et al. [33] used the Histogram of Gradient (HOG) method on MHI, they then trained their method using Support Vector Machines (SVM). The KTH dataset was used to evaluate their method. The method achieved good recognition rates. However, the method did not consider dynamic movements.

Rahman et al. [34] extended the MHI algorithm to solve the occlusion problem. Gradient based optical flow vectors were extracted from each video sequence and split into 4 different direction vectors. From these four direction vectors, Motion History Images are computed. Classification was performed using K-Nearest Neighbours on an aerobic dataset.

Li et al. [35] propose a Multiple Key Motion History Image. Multiple Key MHI were extracted from a video stream. Spatial pyramid matching (SPM) with 2D entropy was then used to obtain spatio-temporal characteristics from the action. The spatial pyramid matching the 2D entropy was combined with spatial pyramid Zernike moment from the Motion History Image Edge (MHIE) to form a feature vector. The method was trained using Support Vector Machines and validated using the KTH dataset.

Murtaza et al. [36] use the Histogram of Gradient (HOG) description method to extract features from MHI to recognize human actions. The features are trained using k-nearest neighbours (KNN). Testing of this approach was done on the MuH AVI-14 and MuHAVI-8 datasets. Further testing on other datasets would need to be considered in order to verify the performance of the system based on other types of activities, actors and video environments.

Murtaza et al. [37] build on their approach of using Histogram of Gradients descriptors on MHI by using a median filter to remove noise from the MHI before applying the Histogram of Gradients feature extractor. The authors test this approach on the MuHAVI-uncut dataset. This approach can be further optimized by finding an optimal tau value or using other classifiers and descriptors such as Support Vector Machines (SVM) and Speed Up and Robust Features (SURF) respectively.

2.3.2 Bag of Visual Words

The Bag of Visual Words (BOVW) model was first introduced by Csurka et al. [38]. The model has been used in various applications such as object recognition, text classification, Natural language processing and human action recognition. The aim of the model is to take a set of images, extract feature descriptors from them, and then build a vocabulary of features using K-Means clustering [38].

Laptev and Lindeberg [39] extended the Harris detector into space–time interest points and detected local structures that have significant local variation in both space and time. The authors then estimate the spatio-temporal extents of the detected events and then computed and extracted descriptors using the scale-invariant Harris descriptor. Using this extension, they classified events and constructed a video representation in terms of labelled space-time points. The representation was then applied to human action recognition using SVM [40]. However, they did not compensate for an extension which would consider the invariance of spatio-temporal descriptors with respect to the direction of motion, changes in image contrast and rotations. This method also only tested the walking activity and the results obtained could be lower for other activities as there was a strong focus on walking.

Niebles and Fei-Fei [41] used a collection of spatial and spatial temporal features extracted from static and dynamic interest points. A hierarchical model was proposed, which was characterized as a constellation of bags-of-features. This enabled them to combine both spatial and spatial-temporal features. Their results were promising, though the lack of large and challenging video datasets to thoroughly test their algorithm poses an interesting topic for future investigation. The authors also did not consider a unified framework by combining generative and discriminative models for human action recognition. For similar actions (e.g., “running” and “walking”), classification may benefit from a discriminative model.

Niebles et al. [42] presented an unsupervised learning method for human action categories. Given a collection of unlabelled videos, they aimed to automatically learn different classes of actions present in the data, and applied the learned model to action categorization and localization in the new video sequences. A video sequence was represented as a collection of spatial-temporal words by extracting space-time interest points. The method automatically learned the probability distributions of the spatial-temporal words corresponding to the human action categories. The learning of probability distributions was achieved by using the probabilistic Latent Semantic Analysis (pLSA) model. Given a video sequence, the model could categorize and localize the human action(s) contained in the video. Due to the nature and complexity of the datasets used for testing, their accuracy was fairly low compared to other methods proposed in literature.

Scovanner et al. [43] introduce a 3-dimension Scale Invariant Feature Transform (SIFT) descriptor for video or 3D imagery. Bag of visual words were used to represent the videos in order to produce a feature vector for training. Training was done using SVM and a leave-one-out cross validation approach. The method was then evaluated using the Weizmann dataset.

Akila et al. [44] combined local and global features to improve action recognition. Speed Up Robust Features (SURF) was extracted as the local features and HOG features were used for the global features. SURF and HOG features were extracted directly from video sequences. The BOVW model was then used to obtain feature vectors from the extracted SURF and HOG features. Training was done using SVM and the method was validated using the KTH dataset.

Foggia et al. [45] proposed a method that modelled each action using a high-level feature vector. The feature vector was computed as a histogram of visual words. The visual words extracted was obtained by analysing the global descriptors of a scene. The proposed method was broken into two levels. The first level extracted MHI, Average Depth Images (ADI) and Difference Depth Images (DDI) from the input video stream. Hu moments were performed on the MHI and ADI images, and the radon transform was used on the DDI images. The extracted features from the Hu moments and radon transform were then clustered to form a codebook. This resultant feature vector is then trained using a Support Vector Machine classifier. The method was validated on the MIVIA [46] and the MHAD datasets [47]. Their method could be further optimized by considering other values of tau for generating the MHI. On lower level semantic datasets such as MHAD, their method did not perform well. Lower level semantic datasets contained actions such as running and walking whereas higher-level semantic datasets contained actions such as eating and drinking.

Shukla et al. [48] used depth maps obtained from the Microsoft Kinect sensor. The Bag of Words model was then used to extract the difference in features across the temporal domain. A temporal Bag of Words model was then used on top of spatial-temporal features. The resultant feature vector was used to train a SVM.

2.3.3 Spatio-temporal analysis using other techniques

Yilmaz and Shah [49] built space–time volumes in (x, y, t) space by utilizing time as the third-dimension. Space–time volumes were matched using features from Poisson equations and geometric surface properties, respectively. A sequence of such 2D contours with respect to time generates a spatio-temporal volume (STV) in (x, y, t) space. This can be treated as a 3D object in the (x, y, t) space. The STV was analysed by using the differential geometric surface properties, such as peaks, pits, valleys and ridges, which are important action descriptors that capture both spatial and temporal properties. The short fall of this method was the presence of sequences containing large velocities where the displacement exceeds the size of the local neighbourhood resulting in the motion feature being un-detected.

Su et al. [50] used Local Binary Patterns to analyse the spatio-temporal pattern of activity width vectors obtained from the 2D representation. The activity width vectors were converted to the grey value successively according to the order in activity sequence and the grey image was formed in spatio-temporal space. In their work, the envelope of the activity and the texture of the silhouette was not considered.

Singh et al. [51] derived directionality-based feature vectors (directional vectors) from silhouette contours and used the distinct data distribution of directional vectors in a vector space for clustering and recognition. The algorithm could handle changes in view angle, scale, background and clothing and was translation independent. It could also deal with limited occlusion of the subject. However, for people with significantly different body shape the algorithm would need to be trained on a completely different training set and would no longer be compatible with the previous training data.

Chen et al. [52] presented a local spatio-temporal descriptor for action recognition from depth video sequences, which was capable of distinguishing similar actions as well as coping with varying speeds of actions. This descriptor was based on three processing stages. In the first stage, the shape and motion cues were captured from a weighted depth sequence by temporally overlapped depth segments, leading to three improved Depth Motion Maps (DMMs). In the second stage, the improved DMMs were partitioned into dense patches, from which the local binary patterns histogram features were extracted to characterize local rotation invariant texture information. In the final stage, a Fisher kernel was used for generating a compact feature representation, which was then combined with a kernel-based extreme learning machine classifier. This method however only achieved a recognition rate of 84% on the MSRDailyActivity3D dataset. This was due to noise and the cluttered backgrounds of the dataset. This was a clear flaw in the method as it did not consider noise and the possibility of the cluttered backgrounds of a dataset which are very evident in a real-world scenario.

Duan-Yu et al. [53] proposed an approach which enclosed a star figure by a bounding convex polygon to effectively and uniquely represented the extremities of the silhouette of a human body. This then enabled the human action to be represented as a sequence of the star figures parameters. The convex polygon-based star figure parameters were represented as Gaussian Mixture Models (GMM). This method however could not accurately recognize skipping, this was because when skipping, one foot is usually kept as high as the knee of the other foot and the foot kept off the ground is usually moving forward and backward near periodically. In this case, this action pattern is highly similar to walking action and thus would be falsely detected.

DeMenthon and Doerman [54] used spatio-temporal descriptors for video retrieval to recognize human actions in surveillance video. The method was tested using actions such as pulling and closing drawers.

2.4 Human Action Recognition using Hidden Markov Models

One of the most popular state space models is the Hidden Markov Model (HMM). In the discrete Hidden Markov Model formalism, the state space is considered to be a finite set of discrete points. The temporal evolution is modelled as a sequence of probabilistic jumps from one discrete state to the other.

One of the earliest approaches to recognizing human actions via HMMs was proposed by Yamato et al. [55]. They recognized tennis shots including, the backhand stroke, backhand volley, forehand stroke and forehand volley etc. Features extracted were formed from time-sequential images. The feature vectors were then converted into a symbol sequence by using vector quantization. HMM's were then used to learn the resultant feature symbols and predict the actions being performed.

Wu et al. [56] performed human activity recognition based on the combination of SVM and HMM. They first used a RGBD sensor (Microsoft Kinect) as the input sensor and extracted a set of the fusion features, including motion, body structure features and joint polar coordinates features. The authors then combined SVM and HMM, which involved SVM characteristics that can reflect the difference between the samples and the HMM characteristics that was suited to deal with the continuous activities.

Xu et al. [57] used the Hidden Markov Models approach to recognize human actions for gesture and intent recognition. Features were extracted using the Fast Fourier Transform (FFT) method. The extracted features were then fed into the HMM for training and classification.

Afsar et al. [58] presented an automatic human detection and action recognition system using Hidden Markov Models and Bag of Words. Background subtraction was performed using the Gaussian Mixture Model (GMM). They reported that the algorithm was able to perform robust detection in a cluttered environment with severe occlusion.

Qu et al. [59] used the Hidden Markov Model and vector quantization algorithms as well as the LBG algorithm to process the codebook for vector quantization. They then process this into an HMM to perform action recognition.

Hoang et al. [60] proposed a human action recognition system that converts every sequence of human gestures into sequences of Skeletal Joint Mapping (SJM). They then assign corresponding observation symbols to each SJM. These observation sequences were then used to train the hidden markov models corresponding to human actions such as walking.

Nie and Ji [61] proposed a novel human action recognition algorithm that was able to capture both global and local dynamics of joint trajectories by combining a Gaussian-Binary Restricted Boltzmann Machine (GB-RBM) with a HMM. They presented a method to use RBM as a generative model for multi-class classification. Experimental results on benchmark datasets demonstrated the capability of the proposed method to exploit dynamic information at different levels. Temporal motion of body joints carries crucial information about human actions. However, current dynamic models typically assume stationary local transition and therefore are limited to local dynamics. The performance of the method was below the method reported in [62]. Such result was reasonable because their method only used a subset of the joints, and did not use any features from the depth images.

Lu and Peng [63] addressed the problem of achieving accurate skeleton information from the segmentation method by depth sensors to obtain depth images, and subsequently the human skeleton. Hidden Markov Models were then used to train the features which were obtained from coordinate transformations. With their proposed method they achieve an 80% recognition rate on the MSR-Action3D dataset. Complex actions such as fights and hand shaking were a challenge for this method.

2.5 Human Action Recognition and Deep learning

Deep learning is growing rapidly in the artificial intelligence and data science fields. It is a subset of machine learning approaches that aims to learn feature hierarchies. Every layer of the feature hierarchy makes obtaining further features an easier problem. Many deep learning approaches to human action recognition have already been proposed in literature [64] [65] [66]. Its use for human action recognition is widely reported in the literature. Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) head the way for deep learning in the human action recognition space.

Wang and Yeh [65] investigated the use of CNNs for feature extraction. Their model was trained using Support Vector Machines (SVM) and K-Nearest Neighbours. Their experiments showed that by using Convolutional Neural Networks (CNN) with classifiers such as Support Vector Machines and K-Nearest Neighbours, better performance was achieved than by just using Convolution Neural Networks by itself. They, however, did not perform their test on the entire dataset and used a leave-one-out-cross validation approach to further validate the robustness of their proposed approach.

Chao li et al. [66] transformed a skeleton sequence into an image and then performed end to end learning of deep CNNs. The generated skeleton-based image contains temporal information. The method was tested on the NTU RGB+D dataset and a pre-trained 5-layer CNN.

Ji et al. [67] use 3D CNNs to recognize human actions. Their proposed model extracted features from both the spatial and temporal dimensions by performing 3D convolutions. The developed model generated multiple channels of information from the input frames, and the final feature representation was obtained by combining information from all channels. The model was tested using the KTH and TRECVID 2008 datasets. The 3D CNN is made up of 1 hardwired layer, 3 convolutional layers, 2 sub-sampling layers and 1 full connection layer. The developed 3D CNN model was trained using a supervised algorithm and required a large number of labelled samples. The system could be further optimized by reducing the number of labelled samples. This can be done by using a pre-trained model from unsupervised algorithms [67].

Li et al. [68] used a skeleton based approach using Long Short-Term Memory (LSTM) and CNN to recognize human actions. They used skeleton sequences as a feature extraction technique and passed these features into the LSTM and CNN network. One of the challenges using CNN based methods was to effectively capture spatio-temporal information of a skeleton sequence using image-based representation. It was inevitable to lose temporal information during the conversion of 3D information into 2D information.

Nair et al. [69] used Generalized Regression Neural Networks (GRNN) to compute the functional mapping from action features to its reduced eigenspace representation. Histogram of Flow (HOF) and Local Binary Patterns (LBP) were combined to formulate a feature vector from a human silhouette. The algorithm however could not distinguish between jump, jumping jack and side motion from the Weizmann dataset. This was due to the fact that these motions are similar in a video.

Zhdanov et al. [70] proposed a CNN-based hierarchical recognition approach that recognizes 20 actions from the kinetics dataset. Their method aims to address the problem of learning actions that have similarities between them for human action recognition. The model built uses a two-level structure to recognizes actions. The first layer takes in a video and extracts the high-level action group using a high-level classifier. This is then fed in to the second layer which is conditioned on the predicted group and is then used to predict the final action [70].

Laptev et al. [71] proposed long-term temporal convolutions for recognizing actions from their full temporal extent. They combined long term convolutions with CNN models. In their work, the impact of different low-level representations such as optical flow vector fields and raw video pixels were studied. To achieve the proposed method, Laptev et al. [71] used a 5-layer CNN network with $3x3x3$ filters. Max pooling and ReLU were used in between all the layers. The inputs into the network were a 2 channel (flow-x, flow-y) or three channels (R, G, B) at different temporal resolutions. The method was then validated on the UCF101 and HMDB51 datasets.

2.6 Human Action Recognition Using Other Methods

Sun et al. [72] introduced the concept of joint self-similarity volume (SSV) for modelling dynamic systems. They showed that by using an optimized rank-1 tensor approximation of joint self-similarity volumes, low dimensional descriptors can be obtained. The method is generic and can be applied to different low-level features such as histograms of gradients and silhouettes. Their method achieved rather good results on 3 different datasets, however the occlusion problem was not addressed and the system will need to be tested against occlusions to verify its performance and robustness.

Sepena et al. [73] applied Dynamic Time Warping (DTW) on feature vectors to recognize human actions. Features were extracted from body part tracking using depth cameras. This technique was used to recover human joints body part information and classification done using K-Nearest Neighbours. The approach was tested only on 4 activities and may not be robust enough for distinguishing for activities such as run and walk, for example.

Wing et al. [74] proposed a data driven approach to selecting the number of visual of words by using the localized generalized error of the Radial Basis Function Neural Network (RBFNN). The approach trained a radial basis function neural network for each bag of word size, with a pre-selected pool of bag of word size. The localized generalized error was then computed for the different bag of words sizes. The approach was validated on the KTH and UCF datasets.

Wang et al. [75] presented a novel descriptor for 3D motion trails. The proposed descriptor, described the motion pattern in the 3D motion trail. The 3D motion trail model represents the movements of the action. A Gabor descriptor was then used to obtain features for training. Classification was done using a kernel extreme learning machine. The method was then evaluated using the MSR Action3D dataset.

2.7 Summary of Performance of HAR Methods from Literature

From Table 1, it can be seen that from the holistic approaches provided in the literature, Bappaditya [15] achieved the highest recognition rate on the Weizmann and KTH datasets. A draw back to approaches found in literature is that they are not tested on all datasets available. By extensively testing the method on different datasets, a given system can be refined to a more robust and generic one.

Table 1 - Comparison of recognition rates in Literature

Name	Accuracy (%) (Weizmann dataset)	Accuracy (%) (KTH dataset)	Accuracy (%) (IXMAS dataset)
Niebles [41]	72.8	n/a	n/a
Zhao [1]	98.7	90.8	n/a
Weinland (MHV) [28]	n/a	n/a	93
Weinland (MEV) [28]	n/a	n/a	80
Bappaditya [15]	100	92	n/a
Nair [69]	93	n/a	n/a
Sconanner [43]	82.6	n/a	n/a
Niebles [42]	90	83.33	n/a

2.8 Conclusion

This chapter has surveyed various techniques and approaches within the human action recognition domain. Popular approaches based on spatio-temporal analysis were reviewed in detail. From the literature, it is evident that many great strides have been taken towards solving the human action recognition problem in the computer vision domain. This dissertation will focus on Motion History Images based on spatial temporal analysis.

CHAPTER 3 – MOTION HISTORY IMAGES AND ITS VARIANTS

3.1 Introduction

This chapter introduces the concept of Motion History Images (MHI) and its variants and provides background into MHI, which is the foundation of the human action recognition proposed method presented in this dissertation. Section 3.2 introduces background subtraction techniques which will be used to obtain silhouettes from action recognition videos. The silhouettes obtained will be required to compute MHI images. Section 3.3 explores the various MHI approaches and how to compute them as well as its limitations and how to choose the tau and delta values. Section 3.4 introduces MEI and discusses why it is not suitable for this work. The contribution of this chapter is to show the various MHI approaches and discuss the importance each of its relevant parameters plays when using MHI for human action recognition.

3.2 Background Subtraction Techniques

Background modelling/subtraction in the human action recognition domain is very important, as this is the first step in detecting and identifying moving objects in a video. In order to obtain an MHI, the silhouette of an image first needs to be obtained. This is achieved by extracting the motion object of the video through various frame differencing techniques. The process of detecting or finding out the moving regions of interest from a video or a sequence of images play a crucial role in many vision applications. In order to recognize a human action from a video, the silhouette of the human needs to be extracted, the following approaches can be used to extract silhouettes from a given video [3]:

- Static background subtraction
- Dynamic background subtraction
- Frame subtraction or frame differencing
- Three-frame subtraction

3.2.1 Static Background Subtraction

Among the approaches mentioned, the static background subtraction method is the easiest. The static background approach assumes that the background is static and the indoor environment is in the presence of constant illumination. Background subtraction is then achieved by subtracting the static background image. Assuming there is only one object in each frame, the moving object can be extracted as the foreground. Given a sequence of image I, a typical background subtraction approach for extracting the foreground, I_{fg} is defined as [3]:

$$I_{fg}(x,y,t) = |\ I(x,y,t) - I_{bg}(x,y,t)\ | \tag{3-1}$$

where,
$I(x,y,t)$: is the grey level at location (x,y) of the current frame t,

$I_{bg}(x,y,t)$: is the grey level at location (x,y) of the static background image at frame t,

$I_{fg}(x,y,t)$: is the grey level at location (x,y) of the extracted foreground at frame t.

3.2.2 Dynamic Background Subtraction

Dynamic background subtraction methods are more difficult to compute. This is because the background first needs to be modelled in order to get the moving object of interest. A smart background model should have the ability to analyse the most recent information about an image sequence or video and continuously update the fast changes of the scene's background. The following background subtraction methods can be used to build a background model:

- Background as the average [76] or median [77] of the previous N frames.
- Background as a running average [3].
- Gaussian Mixture Model (GMM) [78].

3.2.2.1 <u>Background Subtraction using Averaging Filtering</u>

Given a sequence of images I, this type of background subtraction technique takes the mean of the n previous frames and subtracts it from the current frame to obtain the foreground. The average background of the n previous frames, $B(x,y,t)$ at pixel location (x,y) at frame t is defined as [3] [76]:

$$B(x, y, t) = \frac{1}{n} \sum_{i=0}^{n-1} I(x, y, t - i)$$

(3-2)

where, $I(x, y, t - i\ )$: the grey level at position (x, y) at frame $t - \ i$ of the image sequence I. The value of the foreground $I_{fg}(x, y, t)$, at position (x, y) at frame t is then obtained as follows [3]:

$$I_{fg}(x, y, t) = \begin{cases} 1, & |\ I(x, y, t) - B(x, y, t)| > Th \\ 0, & otherwise \end{cases}$$

(3-3)

where Th is the threshold applied to obtain the foreground image.

3.2.2.2 Background Subtraction using Median Filtering

This approach to background subtraction is similar to the mean approach, however instead of taking the mean of the previous n frames, the median is considered. Given a sequence of images I, the median background $B(x, y, t)$ at pixel location x, y at time t is defined as [3] [77]:

$$B(x, y, t) = median_{\ i \in \{0, \dots n-1\}}\ I(x, y, t - i)$$

(3-4)

where $I(x, y, t - i)$ is the grey level at position (x, y) at frame t of the sequence of images I

The value of the foreground $I_{fg}(x, y, t)$ at position (x, y) at frame t is then obtained as follows [3]:

$$I_{fg}(x, y, t) = \begin{cases} 1, & |\ I(x, y, t) - B(x, y, t)| > Th \\ 0, & otherwise \end{cases}$$

(3-5)

where $I(x, y, t)$ is the grey value of the images sequence at time t and Th is threshold applied to obtain the foreground.

3.2.2.3 Background Subtraction as a Running Average

The background running average model at each pixel location is based on the pixel's recent history. The history is the average of the previous n frames or the weighted average where recent frames have higher weight. The weighted average is computed as the chronological average from the pixel's history and no spatial correlation is used between different pixel locations. Considering the background as a running average, the grey value of the background image $I_{bg}(x, y, t)$ at position (x, y) at frame t is then defined as follows [3]:

$$I_{bg}(x,y,t) = \alpha I\ (x,y,t) + (1 - \alpha)\ I_{bg}(x,y,t-1)$$

(3-6)

where, α is the learning rate which is typically a value of 0.05 [3],
$I(x, y, t)$ is the grey level of the current frame t and $I_{bg}(x, y, t-1)$ is the grey level of the previous background image at frame $t - 1$.

3.2.2.4 Gaussian Mixture Model

One of the most famous dynamic background subtraction approaches is the Gaussian mixture model (GMM). It was first introduced by Stauffer et al. [78]. The idea of the GMM approach is to present a pixel based multimodal description of the background in order to eliminate repetitive motion. K Gaussian distributions are used to model the recent history of each pixel, $\{X_1, \ldots, X_t\}$, The probability of observing the current pixel value is defined as follows [78]:

$$P(X_t) = \sum_{i=1}^{K} \omega_{i,t} * \eta(X_t, u_{i,t}, \Sigma_{i,t})$$

(3-7)

where,
K: the number of Gaussian distributions,
$\omega_{i,t}$: an estimate of the weight for the ith Gaussian distribution at time t
$\mu_{i,t}$: the mean value of the ith Gaussian distribution at time t,
$\Sigma_{i,t}$: the covariance matrix of the ith Gaussian distribution at time t,
η: probability density function given as [78]:

$$\eta(x_t, \mu, \Sigma) = \frac{1}{(2\pi)^{\frac{\dim(X)}{2}} |\Sigma|^{\frac{1}{2}}}\ e^{-\frac{1}{2}(x_t - \mu_t)^T \Sigma^{-1}\ (X_t - \mu_t)}$$

(3-8)

The prior weights of the K distributions at time t, $\omega_{k,t}$ are adjusted as follows [78]:

$$\omega_{k,t} = (1-\alpha)\omega_{k,t-1} + \alpha(M_{k,t})$$

(3-9)

where,

α: is the learning rate,

$M_{k,t}$: is 1 for a model that matched and 0 for the remaining models.

The remaining parameters μ_t and σ^2 of the distribution that matches the new observation are then updated as follows [78]:

$$\mu_t = (1-\rho)\mu_{t-1} + \rho X_t$$

(3-10)

$$\sigma^2 = (1-\rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t)$$

(3-11)

where,

σ^2: is the standard deviation and,

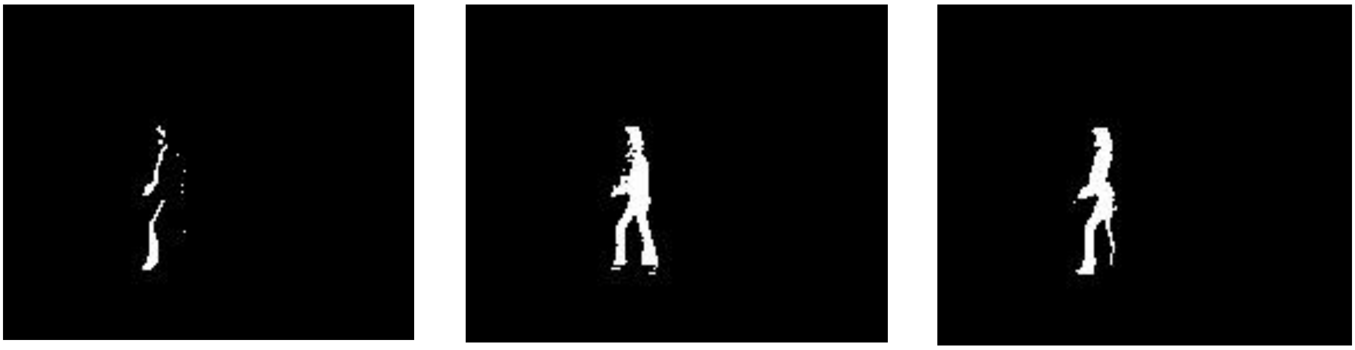
$\rho = \alpha\,\eta(X_t|\mu_k, \sigma_k)$

The first B distributions are then chosen as the background model and is given as [78]:

$$B = argmin_b(\sum_{k=1}^{b} \omega_k > T$$

(3-12)

where, T is a measure of the minimum portion of the data that should be accounted for by the background.



(a) 0.5 **(c) 0.01** **(b) 0.1**

Figure 3.1 - Walking Action with different learning rates for GMM. (a) learning rate of 0.5, (b) learning rate of 0.1, (c) learning rate of 0.01

Examples of silhouettes obtained for different learning rates using GMM can be seen in Figure 3.1 above.

3.2.3 Frame Differencing or Frame Subtraction

In this computationally simple approach, frames are subtracted from each other and the moving regions are accumulated to get the foreground. However, just like other background subtraction methods, this method can suffer from missing information. A typical approach for frame differencing in order to obtain the foreground image I_{fg}, is to compute its grey values, $I_{fg}(x, y, t)$ at pixel location (x, y) at time t as [3]:

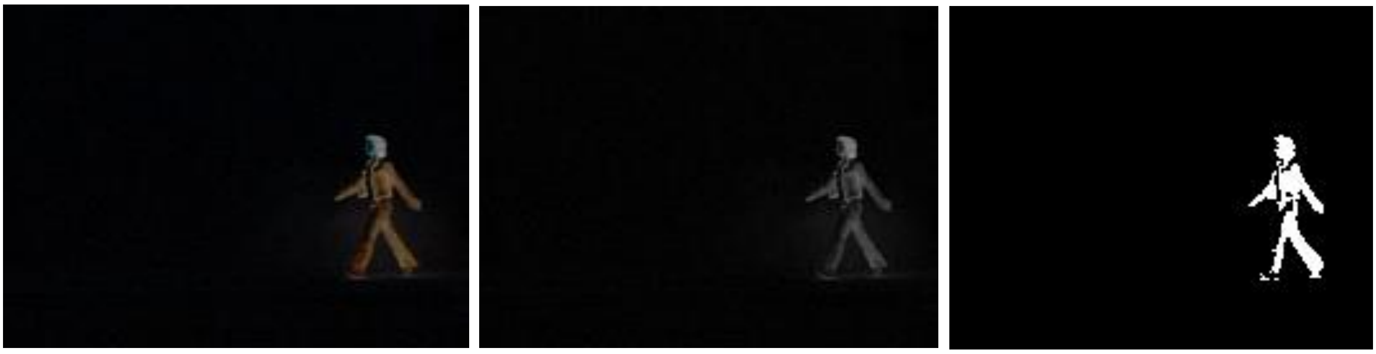
$$I_{fg}(x, y, t) = |\, I(x, y, t) - I(x, y, t-1)| \tag{3-13}$$

where,

$I\ (x, y, t-1)$: is the pixel grey value at time $t-1$,

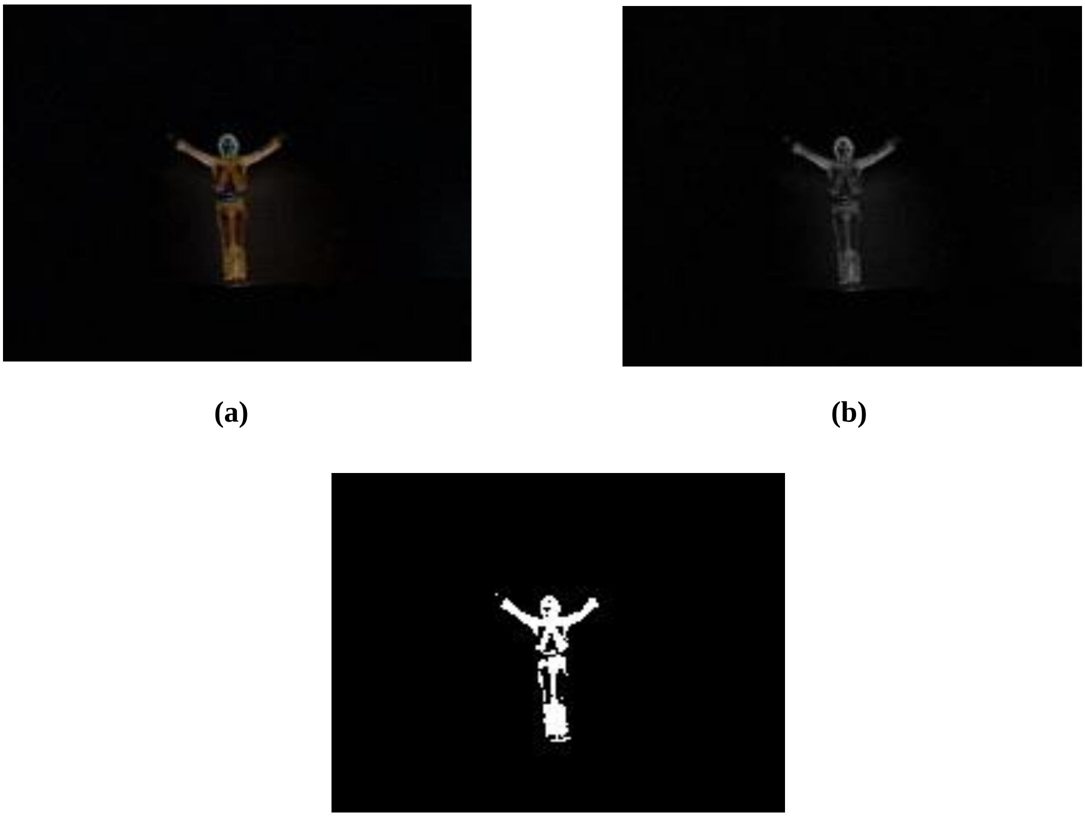
$I\ (x, y, t)$: is the pixel grey value at time t.

Figure 3.2 and Figure 3.3 show the resultant images obtained for the frame differencing technique at a time t.



(a) **(b)** **(c)**

Figure 3.2 - (a) Walking Action at Frame 10, (b) Walking Action at Frame 10 - Grey scale, (c) Walking Action at Frame 10 – Binary Silhouette



(a)

(b)

(c)

Figure 3.3 - (a) Two Hands Waving at frame 10, (b) Two Hands Waving at Frame 10 Grey scale, (c) Two Hands Waving at frame 10 – Binary Silhouette

3.3 Motion History Images

One of the most well-known methods for action representation and recognition are the Motion History Image (MHI) and Motion Energy Image (MEI) approaches. These approaches were first introduced by Bobick and Davis [14] as discussed in chapter 2 of this dissertation. The MHI approach is a view-based temporal template method which is simple but robust in representing movements and is widely employed by various research groups for action recognition, motion analysis and other related applications. MHI is a single image by which moving parts of a video can be identified. This single image representation of the sequence results in the entire motion flow of the video. An MHI shows how motion has occurred for a video or sequence of images. It is a greyscale image where the more recent movements have brighter values.

MHI have the following important features:

- MHI can represent a motion sequence in a compact manner. This results in dominant motion information been preserved where the silhouette sequence is condensed into a grey scale image.
- MHI can be implemented in low illumination conditions where the structure of the person can easily be detected.
- MHI keeps the history of temporal changes at each pixel location. The temporal changes at each pixel location decays over time.

3.3.1 Original Motion History Image

The Motion History Image of a sequence of images I, is the function $H_\tau(x, y, t)$ and is defined as [14]:

$$H_\tau(x,y,t) = \begin{cases} \tau, & if\ \varphi(x,y,t) = 1 \\ \max(0, H_\tau(x,y,t-1) - 1), & otherwise \end{cases}$$

(3-14)

where,

(x, y): is the pixel position

t: is the frame index

τ: is the temporal duration of the MHI

$\varphi(x, y, t)$: signals object motion (or presence) in the current video image, which is the binarization of the difference of frames by considering a threshold β, and is defined as follows [14]:

$$\varphi(x,y,t) = \begin{cases} 1 & if\ D(x,y,t) > \beta \\ 0 & otherwise \end{cases}$$

(3-15)

where $D\ (x, y, t)$: is the absolute value of the silhouette difference between frames t and t – 1, the absolute difference between consecutive frames, $D\ (x, y, t)$ is defined as follows [3] [14]:

$$D(x,y,t) = |\ I\ (x,y,t) - I(x,y,t-1)\ |$$

(3-16)

where,

$I\ (x, y, t)$ is the pixel grey level at frame t,

$I\ (x, y, t - 1)$ is the pixel grey level at frame t – 1.

Algorithm 3.1 below shows how to compute the original MHI from a video stream.

Algorithm 3.1: Original Motion History Images

<u>Inputs:</u>

Video/Image Sequence – Z = $(I_t)_{t=0,1,2\ldots,n}$ and I_t is the $n\ x\ m$ sized image/frame at time t.

Duration Parameter - τ

<u>Outputs:</u>

MHI image – $MHI_t(x,y)$

```
For Each time $t$
  $Bt: = absolute\_difference\ (I_t, I_{t-1})\ >\ threshold$
End For

For Each time $t$
  For Each $pixel\ (x,y)$
    If $Bt\ (x,y) = 1$
      $MHI_t\ (x,y): = \tau$
    Else If $MHI_{t-1}\ \neq\ 0$
      $MHI_t\ (x,y): = MHI_{t-1}(x,y) - 1$
    Else
      $MHI_t\ (x,y): = 0$
    End If
  End For
End For
```

3.3.2 Modified Motion History Image

The Original MHI equation (3-14), as presented in section 3.3.1, does not accommodate for situations where it might be necessary to reduce more pixel values for each frame. In the original equation, the older pixel values are reduced by ‘1’ [27] [79].

We can then define a decay parameter δ which can be ‘1’ or more. The Modified MHI function, $H_\tau(x, y, t)$, can then be written as [27] [79]:

$$H_\tau(x, y, t) = \begin{cases} \tau & if\ \varphi(x, y, t) = 1 \\ max\ (0, H_\tau(x, y, t-1) - \delta) & otherwise \end{cases} \tag{3-17}$$

where,

τ: is the duration,

δ: is the decay parameter,

$\varphi(x, y, t)$: is the binarization of the difference of the frames which shows where the motion is and is defined in equation (3-15) is section 3.3.1.

3.3.3 Timed Motion History Image

Bradski and Davis [26] generalized the basic MHI to directly encode actual time in a floating-point format, called Timed Motion History Image (TMHI). The new silhouettes values are copied in with a floating-point time stamp. The timestamp format thus becomes “seconds, milliseconds”. By using time to encode the silhouettes, the system becomes independent of speed or frame rates. This ensures that a given gesture will cover the same MHI area at different capture rates.

The Timed Motion History Images function, $tMHI_\alpha(x, y)$, can be defined as follows [26]:

$$tMHI_\alpha(x, y) = \begin{cases} \tau & if\ current\ silhoette\ at\ (x, y) \\ 0 & else\ if\ tMHI(x, y) < (\tau - \ \alpha) \end{cases} \tag{3-18}$$

where,

τ: Current timestamp,

α: The maximum time duration constant (typically a few seconds) associated with the template.

(x, y) : is the grey level pixel position

3.3.4 Choosing tau (τ) and Delta (δ)

3.3.4.1 Selection of Tau (τ) for Original MHI

The value of the duration parameter τ can range from 0 – 255 for a grayscale image. When choosing τ, it is vital to choose a suitable value in order to obtain just the right amount of information required for recognizing human actions. Factors such as the number of frames per video may affect the choice of this value. From Figure 3.4 below, it can be seen that there is a lot of noise in the image, this is due to a combination of the τ value and the threshold value from the frame differencing technique. Choosing both these values becomes vital in a recognition system. If the τ value is smaller than the number of frames, then we lose prior information of the action in its MHI. For example, when τ=20 for an action having 40 frames, motion information of the first frame after 20 frames will be lost if the value of the decay parameter (δ) is 1. However, on the other hand, if the temporal duration value is set to a very high value compared to the number of frames (e.g., 250 in the case for an action with 40 frames), then the changes of the pixel values in the MHI template is less significant. Therefore, this point should be considered when producing an MHI. When MHI is used to represent an action as a whole, setting the duration parameter τ is critical. This is not always easy as the duration of different actions as well as different instances of the same action can vary a lot. The problem with the temporal template representation is that actions that occupy the same space at different times cannot be modelled properly as the observed features will overlap and new observations will erase old ones. This problem is solved by fixing τ to give a short-term motion representation and modelling the actions as a sequence of templates.

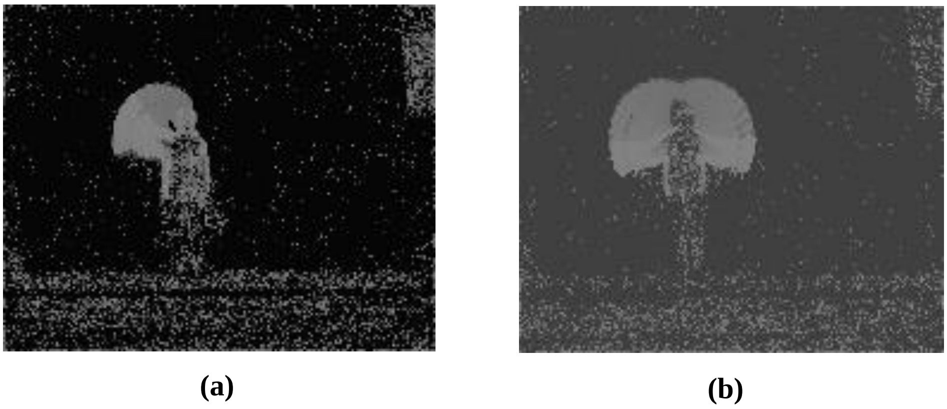


Figure 3.4 - (a) One Hand Waving Action MHI with Noise, (b) Two Hand waving Action MHI with Noise

3.3.4.2 Selection of Delta (δ) for Modified MHI

In the basic MHI method [14], δ is replaced by 1. While retrieving frames, if there is no presence of movement in a specific pixel, where earlier there was movement, the pixel value is reduced by δ. However, having different δ values may provide slightly different information; hence the value can be chosen empirically. When using MHI, users need to consider this parameter as higher values for δ remove the earlier trail of a motion sequence. Experiments of this can be seen in chapter 5 of this dissertation.

It is evident that both the values of τ and δ combined, determine how long it takes for an objects movement to decay to 0, and therefore determines the temporal window size. However, different configurations can lead to the same temporal window (e.g., τ = 10 and δ = 1 leads to the same temporal window as τ = 100 and δ = 10). The combination of a large τ and a small δ yields a slowly-changing continuous gradient, whereas a large τ and large δ gives a more step-like response. An attempt to model τ and choosing δ was done by Rahman et al [3] [80]. This provides an insight into what parameters and design choices one has to make as well as the impact of choosing different parameters or designs when using MHI.

3.3.5 Limitations of the Motion History Image method

Although MHI have many advantages and uses, especially in the action recognition domain, it does have limitations. The search for a method to generically identify actions is a difficult task, and no single method can address the various problems of human action recognition [3]. Some of the short falls of the MHI approach are discussed in the sub-sections below.

3.3.5.1 Motion Overwriting

The first limitation of MHI is the motion overwriting problem. A prime example of this would be the running motion for an MHI. If a person runs from left to right and then right to left, the resultant MHI will have no clue of the fact that the person ran from left to right earlier. From this example, it easy to notice that the resultant MHI will have lost some direction information from the video. This problem can be solved by using multiple cameras or splitting optical flow vectors which will preserve the information [81].

3.3.5.2 <u>Multiple People in a scene</u>

MHI does not account for two people being in the scene doing either the same activity or different activities, a possible solution to this is to use a bounding box around the person to track each individual's movement [3].

3.3.5.3 <u>Camera Movements</u>

Camera movements is another limitation to the MHI method. If you have a person running across the scene and the camera is moving, then the resultant MHI will cover more regions of movement which will result in unwanted information being generated [3].

3.3.5.4 <u>Multiple Actions</u>

MHI cannot differentiate an action when multiple actions are being performed, such as, if a person is waving while running and the action to be recognized is waving hands. This will require some sort of tracking to distinguish the walking action from the waving action [3].

3.4 Motion Energy Images

Motion Energy Images (MEI) is a cumulative binary motion image that can describe where a movement is in the video sequence. It was first introduced by Davis et al [14] as a temporal template for human actions. It is computed from the start frame of the video to the final frame. The moving object's sequence is maintained throughout the action region of the image.

If two consecutive images have some change in it, i.e., have some existence of movement, then the resultant MEI image will have pixel values of '1' for pixels where there is some presence of motion. Similarly, if the next image has any movement, then the additional pixels will be changed from '0' to '1'. Through this manner, a cumulative binary image is constructed that presents the history of the motion presence as a binary image. MEI coarsely describes the spatial distribution of motion energy for a given view of a given action [3]. Figure 3.5 shows an example of MEI obtained at different times of a video.

The Motion Energy Images function, $MEI_t(x, y, t)$, at pixel location (x, y) at time t, can then be calculated by the following mathematical expression [14]:

$$MEI_t(x,y,t) = \bigcup_{i=0}^{\tau} S(x,y,t-i)$$

(3-19)

Where, $S(x,y,t-i)$ is the absolute difference between frames t and $t-1$

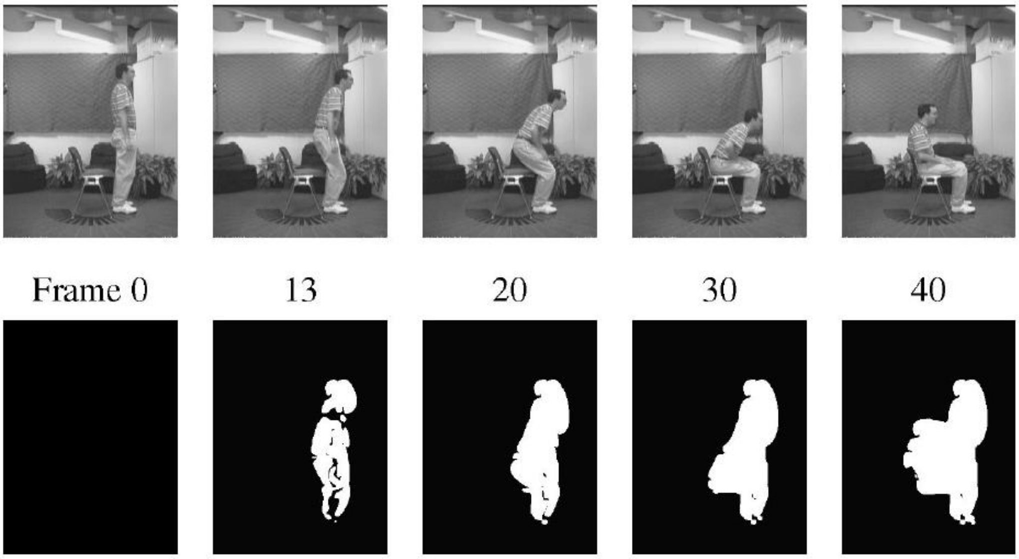


Figure 3.5 - Motion Energy Image of Sitting Down Motion [14]

MEI is a very useful approach when needing to find how the entire motion of an object occurred. It describes an object’s overall pose and shape of motion. Hence its main uses are for template matching in order to recognize an action or objects movement. This however will not be suitable in this work since the MEI is a binary image and extracting features such LBP will have no meaningful value. LBP requires you to threshold the neighbouring pixels in order to obtain the texture of an image, however since the thresholding will take place for binary values where only two values are possible, the LBP code of the image will have no meaningful information about the direction of motion. This was seen in Zhao et al [1] work where they used LBP to extract features from the MEI and was later discarded as it lowered the recognition rates considerably.

3.5 Conclusion

This chapter introduced the concept of MHI and how it can be computed. The theory behind the method was presented, including some common variants of the MHI method. Background subtraction methods were explored in order to obtain silhouettes needed for the MHI algorithm. It was demonstrated how frame differencing techniques is used to obtain the silhouette's in order to compute an MHI in this dissertation. As discussed earlier, MEI is not suitable for extracting features that will be needed for our work. MHI has been chosen in this work as it is simple to implement and is computationally efficient.

CHAPTER 4 – PROPOSED METHOD

4.1 Introduction

In this chapter, the proposed method – MHI using Bag of Visual Words (BOVW) is investigated and is introduced in section 4.7. Section 4.2 introduces the LBP approach and how to compute it. Section 4.3 presents the use of MHI with LBP which, is a common approach in literature for action recognition. Section 4.4 introduces the Scale Invariant Feature Transform (SIFT). In section 4.5 the Speed Up Robust Features (SURF) is introduced. SIFT and SURF are commonly used for the BOVW model. Section 4.6 introduces the BOVW model. It is shown why the use of LBP is not an efficient approach for the human action recognition problem when using MHI.

4.1.1 Proposed Method: Motion History Images using Bag of Visual Words

In this work MHI and BOVW are combined to solve the human action recognition problem. Figure 4.1 shows the overall process of human action recognition for this work. MHI is first generated from an action scene (video or sequences of images). Based on the work done in [38] on BOVW, the SURF method is then used for detecting features, descriptors are then extracted and clustered using K-Means. At the K-Means clustering step, a feature vector is produced. Feature vectors produced are fed into an SVM classifier for training. Testing is done by inputting new action samples which was not trained in order to evaluate the proposed approach. The results achieved from this approach is then compared to the MHI using LBP method and to the state-of-the-art

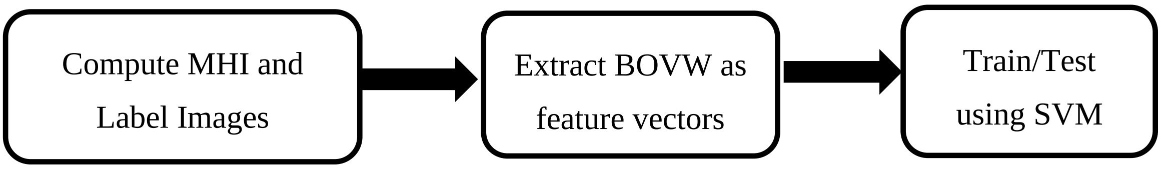


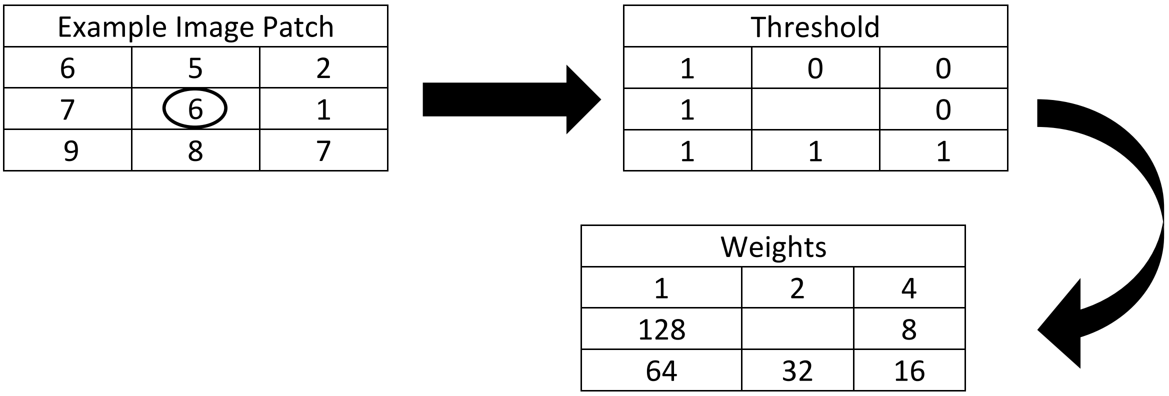
Figure 4.1 – MHI using BOVW – Proposed Method System Design

4.2 Local Binary Patterns (LBP)

LBP is a type of visual descriptor used for feature extraction in the image processing and computer vision domain [82] [83]. It is a simple, yet very efficient texture operator which labels the pixels of an image by thresholding the $3x3$ neighbourhood of each pixel using the centre value, thus resulting in a binary number. The resultant binary number is then translated in to a histogram [84] [85] to form a feature vector. These labels or their statistics, most commonly the histogram, are then used for image analysis. The most widely used versions of the operator are designed for monochrome still images but it has been extended also for color (multi-channel) images as well as videos and volumetric data. Its applications include, texture classification, facial recognition and human action recognition.

4.2.1 Basic Local Binary Patterns

Ojala et al. [82] first introduced LBP to measure local image contrast. The LBP operator worked with the eight neighbours of a pixel by using the centre pixel value as a threshold. If the centre pixel is greater than the neighbouring pixel, write a “0” else write a “1”. Binary numbers are then achieved by concatenating the binary codes in a clockwise direction, which starts from the top-left of the current image patch. The coinciding decimal value is then used for labelling the image [86]. An illustration of this can be seen in Figure 4.2 below.



Example Image Patch		
6	5	2
7	6	1
9	8	7

Threshold		
1	0	0
1		0
1	1	1

Weights		
1	2	4
128		8
64	32	16

Figure 4.2 - Example of Local Binary Pattern, 3x3 Neighbourhood

Binary number = 11110001

Decimal value = 1 + 16 + 32 + 64 + 128 = 241

After applying the LBP operator on the image, a feature vector can be obtained by computing the histogram over the image. The histogram can be seen as a 256-dimensional vector.

4.2.2 Circular/Extended Local Binary Patterns

One limitation of the basic LBP operator is that its small $3x3$ neighbourhood cannot capture dominant features with large scale structures. To deal with this, Ojala et al. [87] extended the basic LBP approach into a circular symmetric representation as can be seen in Figure 4.3. A local neighbourhood is defined as a set of sampling points evenly spaced on a circle which is centred at the pixel to be labelled, and the sampling points that do not fall within the pixels are interpolated using bilinear interpolation, thus allowing for any radius and any number of sampling points in the neighbourhood. The extended/circular $LBP_{P,R}$ can be expressed as follows [87]:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s\left(g_p - g_c\right)2^p \tag{4-1}$$

where,

$g_c = I(x_c, y_c)$: is the grey value of the centre pixel of a local neighbourhood

$g_p(p = 0, \dots, p-1)$: Are the grey values of (x_p, y_p) equally spaced pixels on a circle of radius R (R > 0) that form a circularly symmetric set of neighbours.

And $s\left(g_p - g_c\right) = s(x)$ is defined as [87]:

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \tag{4-2}$$

$g_p(P = 0, \dots, P-1) = I(x_p, y_p)$.

Where x_p and y_p is defined as [87]:

$$x_p = x_c + Rcos(\frac{2\pi p}{P}) \tag{4-3}$$

$$y_p = y_c - RSin(\frac{2\pi p}{P}) \tag{4-4}$$

By the definition from equation 4-1, the extended LBP operator is invariant to monotonic grey-scale transformations preserving pixel intensity order in the local neighbourhoods. The operator LBP (P, R) produces 2^p different output values, corresponding to 2^p different binary patterns formed by P pixels. If the image is rotated, these surrounding pixels in each neighbourhood will move correspondingly along the perimeter of the circle, resulting in a different LBP value, except patterns with only 1s and 0s. Algorithm 4.1 shows how to compute the extended Local Binary Pattern approach.

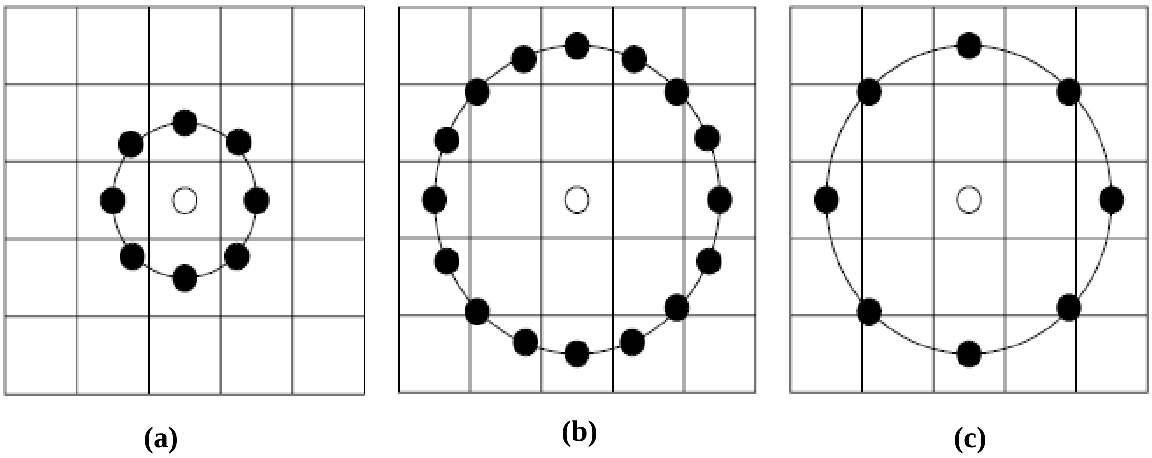


Figure 4.3 - (a) R = 1; P = 8, (b) R = 2; P = 16, (c) R = 2; P = 8 [85]

Algorithm 4.1: Circular Local Binary Patterns Algorithm

Inputs:

Image I – I is an $n \ x \ m$ sized image

Radius – R

Number of sampling points – P

Outputs:

LBP feature vector/histogram V – V is $1 \ x \ n$ size vector

```
For Each Neighborhood
    For Each pixel $(x, y)$ in $I$
        Compute the intensity difference of the current pixel $(x, y)$ with the $P$ neighboring pixels.
        Threshold the intensity difference in order for the negative differences are set to 0 and all positive differences are set to 1, forming a bit vector $B$.
        Convert the $B$-bit vector to its corresponding decimal value and replace the intensity value at $I \ (x, y)$ with this decimal value.
    End For
End For
$V$ ← compute the histogram for the LBP code Image $I$ and store in $V$
```

4.3 Motion History Images Using Local Binary Patterns

In this proposed approach, LBP are extracted from the generated MHI as feature vectors. It is then trained using K-Nearest Neighbours (KNN) and Support Vector Machines (SVM). A high-level view of this design can be seen in Figure 4.4 and a detailed description of the computation can be seen in Algorithm 4.2. The local binary pattern codes extracted from the MHI, encode information about the direction of motion. By using the extended LBP approach, large scale structures can be preserved. his approach is computationally simple and efficient. However, it should be noted that the outer edges of the MHI may be misleading as there is no useful motion information when using the LBP method. LBP also does not accommodate for scale, rotational and translation invariance which will be key to obtaining descriptive features from an action extracted by an MHI, which could lead to low recognition rates.

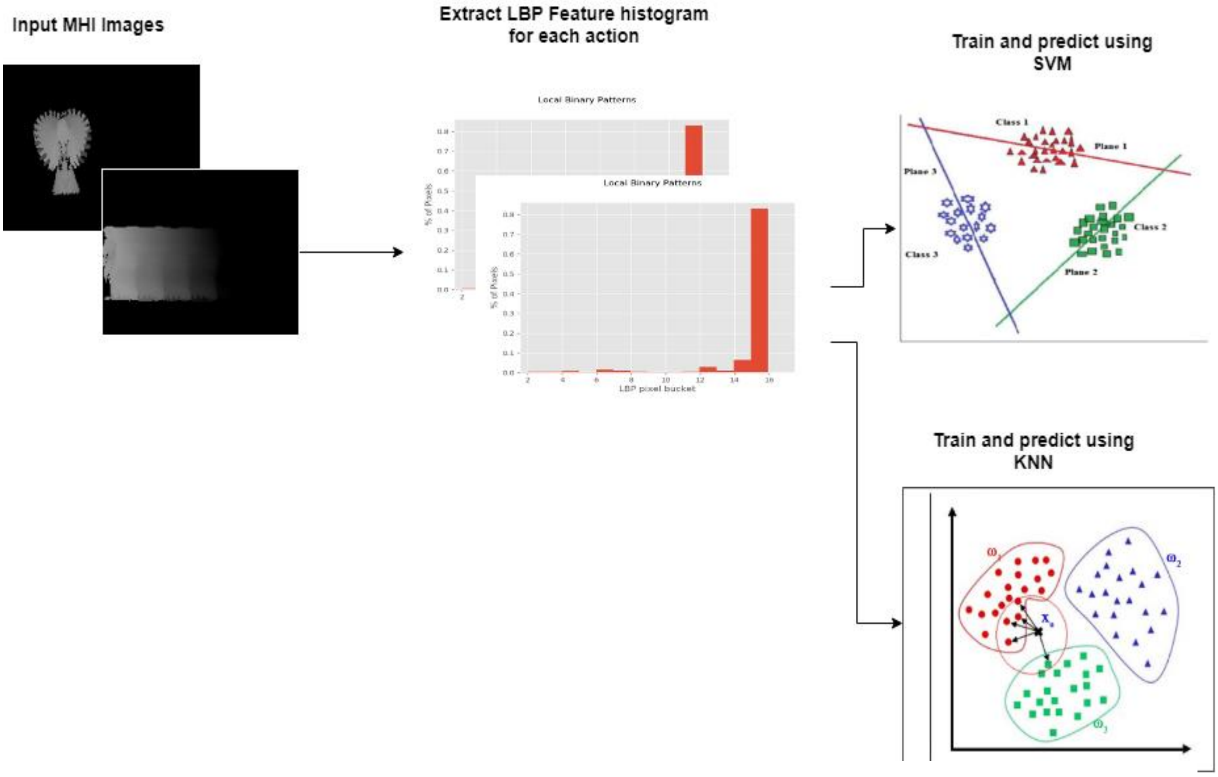


Figure 4.4 - MHI using Local Binary Patterns

In the Local Binary Pattern step, a radius of 1 is used together with 4 sampling points. A bin size of 16 is used to obtain the feature vector. This is then fed into an SVM and KNN classifier for training. An example of the histogram extracted for the walking action of an MHI can be seen in Figure 4.5.

Algorithm 4.2: MHI using Local Binary Patterns

```
Inputs:
Sequence of Motion History Images – $M$ where $M$ is $n\ x\ m$ sized image.
Radius – $R$
Sampling Points - $P$

Outputs:
Feature Vector – $V$ where $V$ is a $1\ x\ n$ vector
1: Initialize variables
        $R = 1$
        $P = 4$
2: Split training and testing sets
3: For each $M$ in training set
        $V \rightarrow\ Extract\ LBP\ features$
        $label \rightarrow\ Assign\ Label\ for\ current\ M\ action$
   End for
4: Train feature vector $V$ with label $L$ using SVM/KNN
```

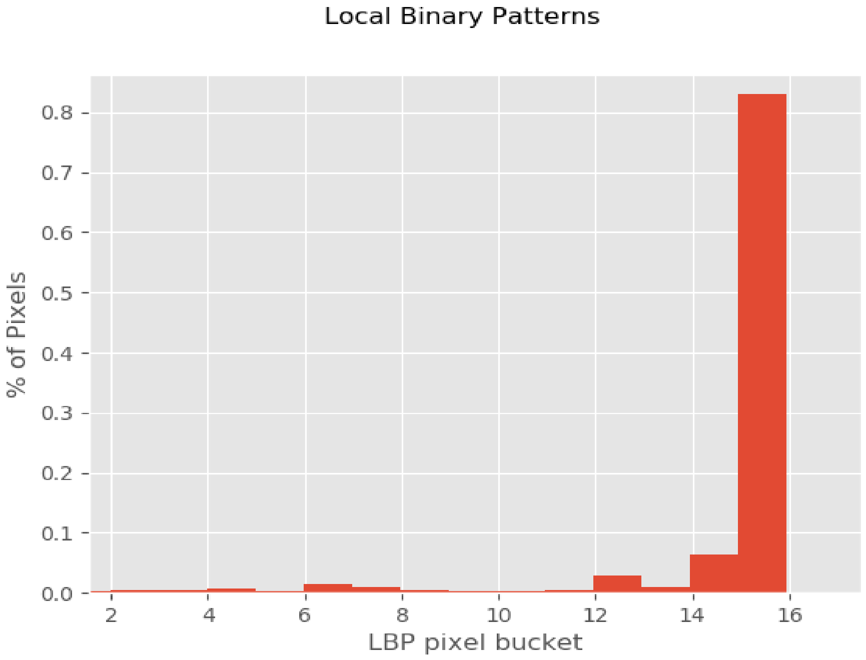


Figure 4.5 - Local Binary Pattern Feature Vector for Walking Action

4.4 Scale Invariant Feature Transform (SIFT)

Scale Invariant Feature Transform (SIFT) was first introduced by Lowe [88] [89] to recognise objects. SIFT is a feature detection algorithm used to detect interest points in an image. The features obtained from SIFT are invariant to image rotation, scaling and translation. The transform is also partially invariant to illumination changes and affine or 3D changes [88]. Its applications include; object recognition, gesture recognition, image stitching and video tracking to name a few. SIFT has been used in human action recognition in various works [43] [90].

4.4.1 SIFT Algorithm

The main stages of the SIFT algorithm is as follows:

- Scale-space extrema detection
- Key point localization
- Orientation Assignment
- Key point Descriptor

4.4.1.1 Scale-space extrema Detection

The first step of the process is to detect points of interest or key points. In this step the difference of Gaussian (DoG) is first computed. The difference of Gaussian (DoG) method is an approximation of the Laplacian of Gaussian (LoG). This is done by using convolution on the input image using Gaussian filters at different scales. In order to get the key points, the maxima/minima of the DoG are taken at different scales.

The approximation of LoG by DoG can be obtained as follows [89]:

$$\sigma\nabla^2 G = \frac{\partial G}{\partial \sigma} = \frac{G(x,y,k\sigma) - G(x,y,\sigma)}{k\sigma - \sigma} \tag{4-5}$$

This results in the Difference of Gaussian as an approximation of the Laplacian of Gaussian as [89]:

$$G(x,y,k\sigma) - G(x,y,\sigma) \approx (k-1)\sigma^2\nabla^2 G \tag{4-6}$$

Where $G(x, y, \sigma)$ is defined as [89],

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}} \tag{4-7}$$

x, y is the pixel location and σ is the standard deviation and k is the scaling factor.
Typical values: k = √2 , $\sigma = 1.6$,

The Difference of Gaussian, $D(x, y, \sigma)$ can then be defined as [89]:

$$\begin{aligned} D(x, y, \sigma) &= \big(G(x, y, k\sigma) - G(x, y, \sigma)\big) * I(x, y) \\ &= L\ (x, y, k_i\sigma) - L(x, y, \sigma) \end{aligned} \tag{4-8}$$

where,
$L(x, y, k\sigma)$ is the convolution of the original image $I(x, y)$ and the Gaussian blur $G(x, y, k\sigma)$ at a scale $k\sigma$.

And the convolution of the original image, $I(x, y)$ and the Gaussian blur $G(x, y, k\sigma)$, $L\ (x, y, k\sigma\ )$ is then defined as [89]:

$$L(x, y, k\sigma) = I(x, y) * G(x, y, k\sigma) \tag{4-9}$$

Noting also that the difference between scales $k\sigma$ and σ is basically the difference of the Gaussian blur images. Figure 4.6 shows the DoG at different scales for an Image.

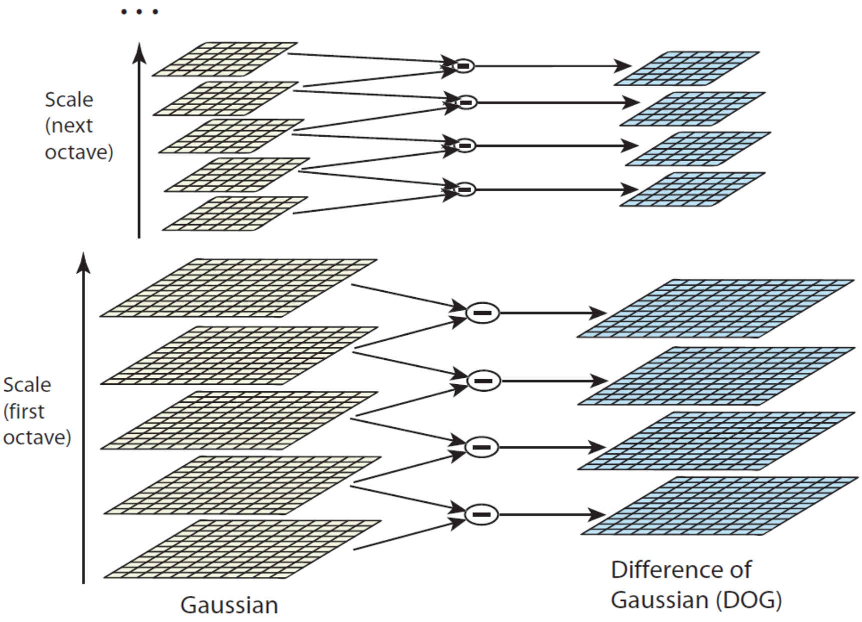


Figure 4.6 - The input image is convolved continuously with Gaussians at each octave in the scale space. The difference of the Gaussians is then computed by subtracting the adjacent Gaussians. [69]

4.4.1.2 Key point localization

Since the scale space extrema detection step produces too many key point candidates, the key point localization step performs a detailed fit to the nearby data for accurate location and scale information. This step rejects low contrast information which is sensitive to noise. In order to accurately determine a key point location, interpolation is used on the nearby data. Interpolation is done using the quadratic Taylor expansion of the Difference of Gaussians scale space function D (x, y, σ) with the candidate key point at the origin.

The Taylor expansion, $D(x)$ is then defined as [89]:

$$D(x) = D + \frac{\partial D^T}{\partial x} x + \frac{1}{2} x^T \frac{\partial^2 D}{\partial x^2} x \tag{4-10}$$

where,

D is the Gaussian Scale space function, D (x, y, σ) and its derivatives are evaluated at the candidate point, and $x = (x, y, \sigma)^T$ is the offset from the sample/candidate point.

In order to eliminate edge responses, the principle curvature of the edge is determined using a hessian matrix, H. This is given as [89]:

$$H = \begin{bmatrix} D_{xx} & D_{yx} \\ D_{xy} & D_{yy} \end{bmatrix}$$

(4-11)

4.4.1.3 <u>Orientation Assignment</u>

In order to obtain invariance to image rotation, each key point is assigned a consistent orientation based on local image properties. For an image sample, $L(x, y)$ at scale σ, the gradient magnitude, $m(x, y)$ and orientation, $\theta(x, y)$ are precomputed using pixel differences and is defined below as [89]:

$$m(x,y) = \sqrt{\left(L(x+1,y) - L(x-1,y)\right)^2 + \left(L(x,y+1) - L(x,y-1)\right)^2}$$

(4-12)

$$\theta\ (x,y) = atan\left(\frac{L(x,y+1) - L(x,y-1)}{L(x+1,y) - L(x-1,y)}\right)$$

(4-13)

Within a region of the key point, an orientation histogram is formed from the gradient orientations. This orientation histogram has 36 bins covering the 360-degree range of orientations. Every sample added to the histogram is weighted by a Gaussian weighted circular window and by its gradient magnitude. The Gaussian weighted window is weighted with a σ that is 1.5 time the scale of the key point.

4.4.1.4 <u>Key point Descriptor</u>

In the descriptor step, the gradient magnitude and orientation at each image sample point is computed in a region around the key point location. This can be seen in Figure 4.7 below obtained from [89]. The created key point descriptors are then weighted by a Gaussian window, which can be seen by the overlaid circle in Figure 4.7.

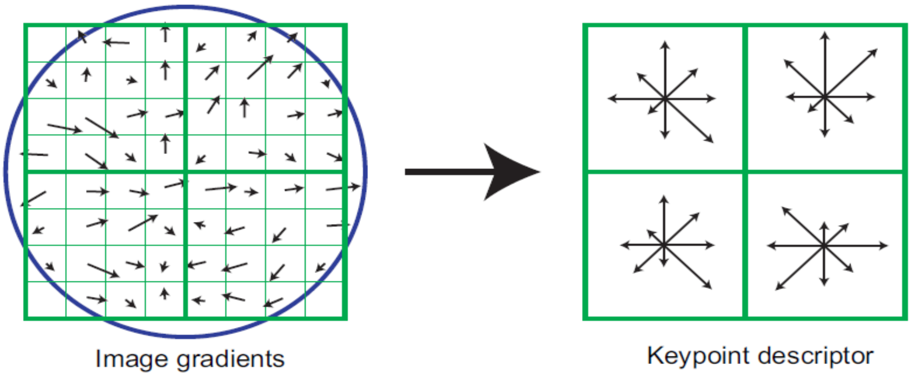


Figure 4.7 - Key Point Descriptor computed from Image Gradients [89]

Algorithm 4.3 below shows how to compute the SIFT detector in order to obtain feature descriptors.

Algorithm 4.3: SIFT Detection and Description [91]

```
Input:
An nxm Image – I
Number of Octaves – num_octaves
Output:
Vector of key points Kp

1: Initialization
        Resample image to double size
2: For each num_octaves
        Generate Gaussian blur intervals
        Generate DoG intervals
        Compute edges for each interval
   End For
3: Search octaves that have extrema stability
4: Generate key points at dominant orientations of extrema
5: For each key point
        Rotate sample grid to key point orientation
        Sample Region and create descriptor
   End For
6:  Save Descriptors in Kp
```

4.5 Speed Up Robust Features (SURF)

Speed Up Robust Features (SURF) is a local feature detector and descriptor. It is used in various applications such as object recognition, classification and image registration. It was introduced by Bay et al. [92] as an improvement to the SIFT method. It is computationally faster than SIFT, hence the name Speed Up Robust Features.

4.5.1 SURF Detection

Squared shaped filters are used as an approximation of Gaussian smoothing, which is done instead of using the Difference of Gaussian approach, as used in the SIFT method. Square shaped filters are faster when using integral images for image convolutions.

Given an Image $I\ (x, y)$, an integral image, $S(x, y)$, at pixel location x, y, can be defined as [92]:

$$S(x,y) = \sum_{i=0}^{x}\sum_{j=0}^{y} I(x,y) \tag{4-14}$$

The integral image thus represents the sum of all pixels of the input image I within a rectangular area. In order to find the points of interest, a blob detector based on the hessian matrix is used. The determinant of the Hessian matrix measures the local changes around a point. Points for which the determinant of the Hessian matrix is at a maximum are chosen. Scale selection is also returned by the determinant.

Given a point $p\ =\ (x, y)$ of an image I, the Hessian matrix $H(p, \sigma)$ at the point p and scale σ is defined as [92]:

$$H(p,\sigma) = \begin{pmatrix} L_{xx}(p,\sigma) & L_{xy}(p,\sigma) \\ L_{xy}(p,\sigma) & L_{yy}(p,\sigma) \end{pmatrix} \tag{4-15}$$

where,

e.g. $L_{xy}(p, \sigma)$ is the convolution of the second order derivative $\frac{\partial^2}{\partial_x \partial_y} g(\sigma)$ of the Gaussian with the image $I\ (x, y)$ at the point x.

A box filter size of 9x9 is used as an approximation of a Gaussian with $\sigma = 1.2$, this represents the highest spatial resolution.

4.5.1.1 Scale space representation

Scale spaces are obtained by applying box filters of different sizes. Bay et al. [92] divided the scale space into octaves. The octaves represent a series of filter response maps. These maps are obtained by convolving the same input image with a filter that increases in size. Instead of iteratively reducing the image size, the scale space is analysed by upscaling the filter size.

4.5.1.2 Interest point localization

Bay et al. [92] apply a non-maximum suppression to a $3x3x3$ neighbourhood to localise interest points in the image and over the scales. The determinant of the Hessian matrix is then computed and interpolated in scale space.

4.5.2 SURF Descriptor

The goal of any descriptor is to uniquely describe an image feature robustly. The SURF descriptor describes the intensity content distribution within an interest point neighbourhood [92]. First order Haar wavelet responses in the x and y direction are computed by exploiting the integral images and thus produces a feature vector of 64 dimensions. This is different from the SIFT method as the SIFT algorithm use gradient information to describe the detected features.

4.5.2.1 Orientation Assignment

In order to obtain rotation invariance, Haar wavelet responses are used in the horizontal and vertical for a circular neighbourhood of 6s around the point of interest, where s is the scale at which the point of interest was detected. i.e. $s = 1.2$. After the wavelet responses are calculated with the Gaussian ($\sigma = 2s$) around the point of interest, they are then represented as points in a space. The superior orientation is calculated by summing all the responses within a sliding window of size $\frac{\pi}{3}$. This can be seen in Figure 4.8 below space. The superior orientation is calculated by summing all the responses within a sliding window of size $\frac{\pi}{3}$. This can be seen in Figure 4.8 below.

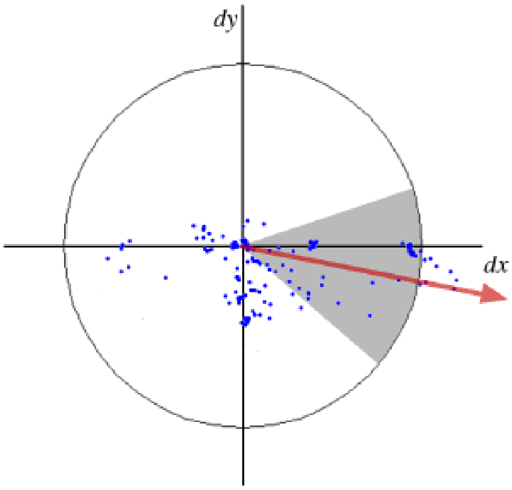


Figure 4.8 - Sliding window orientation of size π/3 [92].

The computation for SURF can be seen in Algorithm 4.4 below.

Algorithm 4.4: Speed up Robust Features (SURF) Algorithm

```
Input:
An nxm Image – I

Output:
Feature descriptor - Kp

1: Detect Interest Points (I)
2: For each I in each Activity
      Compute Integral Image for Image I
  End For
3: For each Octave
    Compute Hessian Matrix(I)
    Kp – Generate Key points from determinant of Hessian Matrix
    End For
4: Compute Descriptor (Kp)
5: Calculate Haar wavelet responses
6: Calculate dominant orientation of an interest point
7: Extract feature descriptor
```

4.6 Bag of Visual Words

The Bag of Visual words (BOVW) model has become one of the more popular methods for image classification, however in the human action recognition domain, it is becoming increasing popular as seen in related works presented in chapter 2 of this dissertation. Csurka et al. [38] proposed a novel method for generic visual categorization by obtaining bag of key points based on a vector quantization of affine invariant descriptors of image patches. The aim of the model is to take a set of images, extract features from them, and then build a vocabulary of features using K-Means clustering. Many approaches for the BOVW model use the SIFT feature descriptor or HOG descriptors for extracting key points. In this work, the SURF descriptor is used with the BOVW model. SURF features as seen in section 4.5 is computationally faster than SIFT and is scale, rotational and translation invariant. This forms part of the main contribution of this work.

The method to obtain the Bag of Features can be described as follows:

- Description and detection of image patches for a given set of labelled training images is conducted.
- A visual vocabulary is then constructed. Each cluster centre is a code vector (visual word of the visual vocabulary), with respect to which descriptors are vector quantized.
- Bags of key points are then extracted for the vocabularies. The bag of key points is trained using Support Vector Machines.

The Bag of Visual Words approach proposed by Csurka et al [38] using the SURF extractor can be seen in Figure 4.9 below.

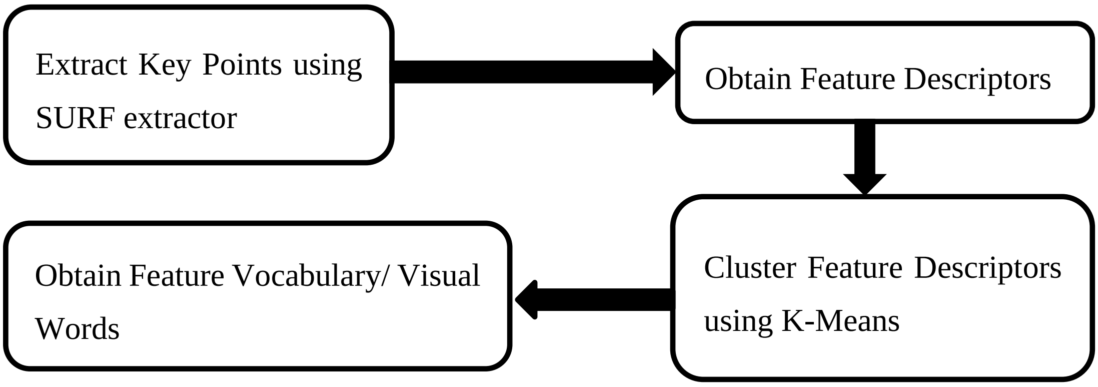
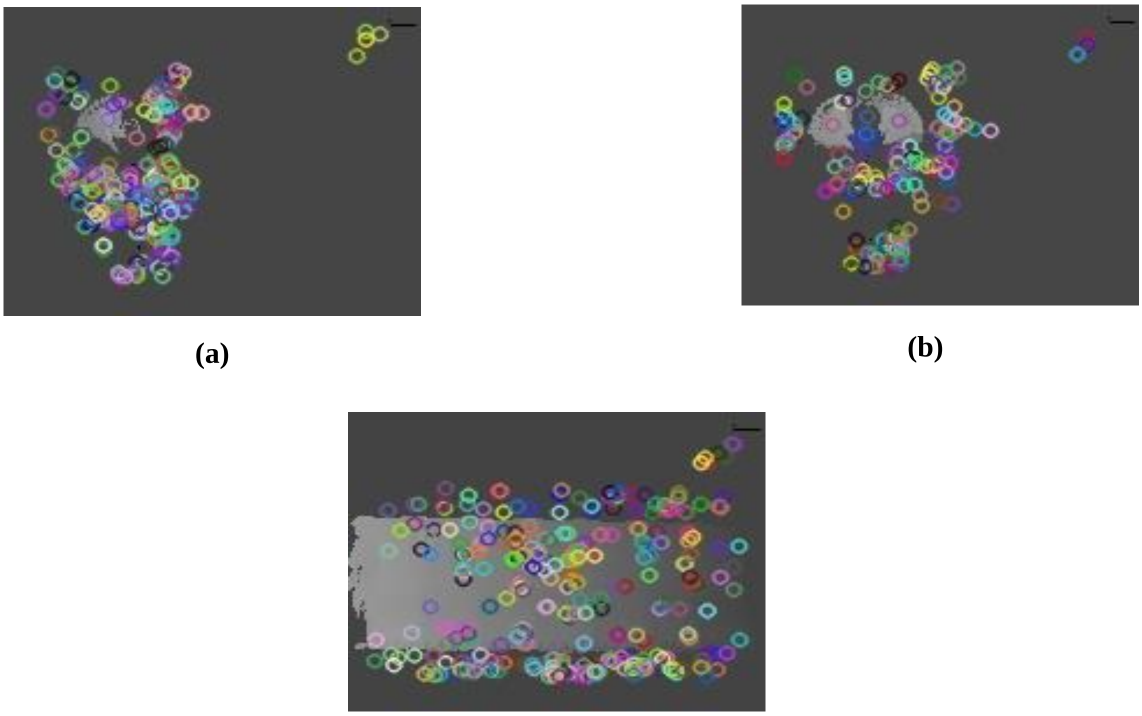

Extract Key Points using SURF extractor
Obtain Feature Descriptors
Cluster Feature Descriptors using K-Means
Obtain Feature Vocabulary/ Visual Words

Figure 4.9 - Bag of Visual Words Algorithm

4.6.1 Speed Up Robust Features (SURF)

In the BOVW approach, the SURF method has two main steps, these are to detect interest points, and then describe them. The descriptors are then fed into the K-Means clustering algorithm to cluster each of them, resulting in a visual word vocabulary. SURF features serve as a shape descriptor for the MHI image that is scale, view and rotational invariant. Figure 4.10 shows SURF features detected for actions obtained from the MHI images.



(a)

(b)

(c)

Figure 4.10 - (a) SURF Key Points for one Hand Waving MHI, (b) SURF Key Points for two Hand Waving MHI, (c) SURF Key Points for Walking MHI

4.6.2 K-Means Clustering

K-Means is a clustering algorithm that partitions n observations in K clusters where each observation belongs to the cluster to which the mean value of elements is the closest to [93]. The pseudocode of this clustering method is shown in Algorithm 4.5. K-Means is an optimization problem, where the objective function is defined as [93]:

$$\underset{S}{argmin} \sum_{i=1}^{k} \sum_{x \in S_i} ||x - c_i||^2 \tag{4-16}$$

where,

x: is an element of the set of observations $\{x_1, x_2, \dots x_n\}$ and each of these observations is a d – dimensional vector,

c_i: is the ith cluster,

K-Means aims to partition the n observations in to $k \leq n$ sets S, and $S = \{S_1, S_2, \dots, S_k\}$

In the K-Means clustering step, the key points are the observations that are clustered to form a vocabulary containing training features. The effects of the K-Means clustering method from Algorithm 4.5 can be seen in Figure 4.11.

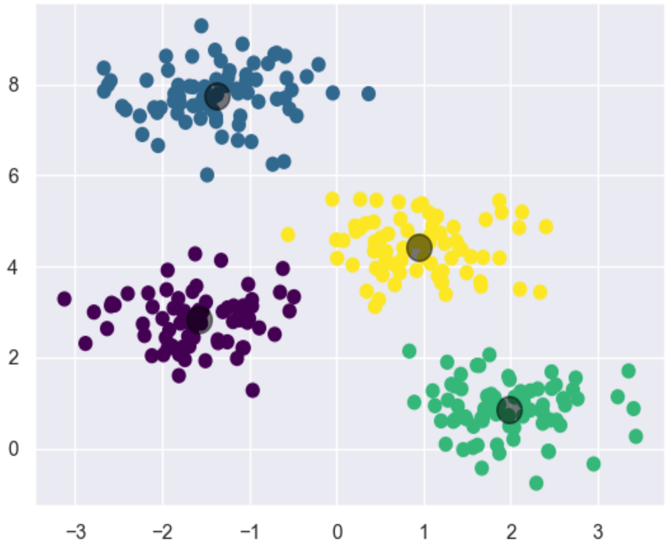


Figure 4.11 - Example of clustering centres that represents the key points from the SURF step to obtain a visual vocabulary

Algorithm 4.5: K-Means Clustering [94]

```
Input:
K (number of clusters)
$x = \{x_1, x_2, \ldots x_n\}$ (set of observations to be cluster
$MaxIters$ (limit of iterations)
Output:
$C = \{c_1, c_2, \ldots, c_k\}$ – Set of Clustered centroids
$L = \{l(x) \mid x = 1,2, .. n\}$ – Set of clustered labels of x

1: Initialize: cluster centroids $c_1, c_2, \ldots, c_k \epsilon R^n$ randomly
        For each $c_i \in C$ do
                $c_i \leftarrow x_j \in X$ – (Random selection)
        End For
        For each $x_i \in X$ do
                $l(x_i) \leftarrow argminDistance\big(x_i, c_j\big) j \epsilon \{1, \ldots, k\}$
        End For
        $changed \leftarrow false$
        $iter \leftarrow 0$
2: For each $c_i \epsilon C$ do
        $UpdateCluster(c_i)$
   End For
3: For Each $x_i \epsilon X$ do
        $minDist \leftarrow argminDistance\big(x_i, c_j\big) j \epsilon \{1, \ldots k\}$
        If $minDist \neq l(x_i)$ then
                $l(x_i) \leftarrow minDist$
                $changed \leftarrow true$
        End If
    End For
4: Repeat 2 and 3 until $change = true$ and $iter \leq MaxIters$
5: Return Set of Clustered Centers C, Set of Clustered Labels L
```

4.7 Proposed Method – MHI using Bag of Visual Words

This section introduces the proposed method for solving the human action recognition problem and for testing the effects that the relevant parameters of MHI has on the overall recognition rate. The main contribution of this work is proposing a new method to solve the human action recognition problem by combing MHI with the Bag of Visual Words (BOVW) method. This approach takes on the ideas of MHI and the BOVW approach and proposes to combine them to solve the human action recognition problem. For each frame of a video, the MHI for each activity is computed. The resultant image is then labelled according to the action being performed. The BOVW extraction method is then used to obtain features. Training is done using the SVM classifier. An overview of the proposed design can be seen in Figure 4.12 and a detailed description of the computation can be seen in Algorithm 4.6. In the BOVW step, the SURF descriptor is chosen instead of the SIFT approach as it is computationally faster.

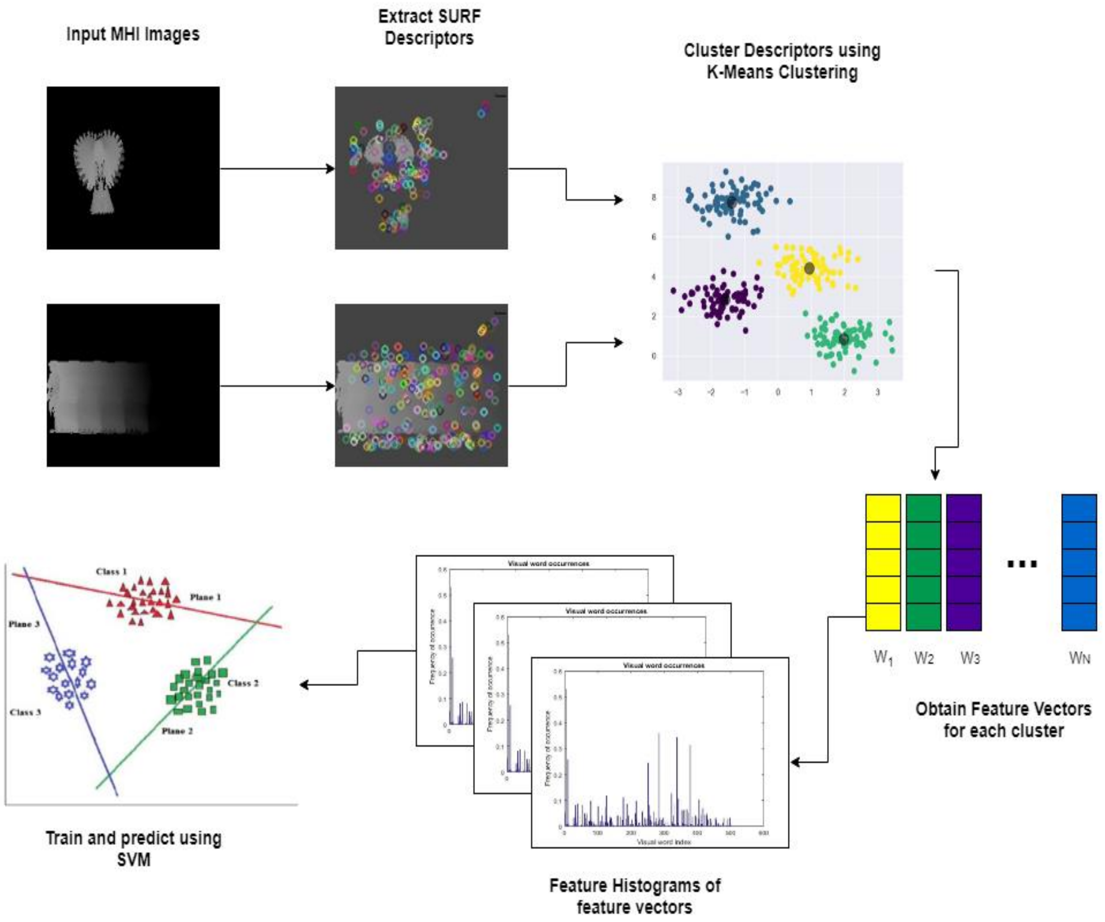


Figure 4.12 - MHI using Bag of Visual Words - Proposed Method

In the proposed approach, SURF features are extracted with a grid step of [8 8] and a Block Width of [32 64 96 128]. In the K-Means clustering step, 500 clusters are used to create a visual word vocabulary. The resultant feature vector obtained can be seen in Figure 4.13 below:

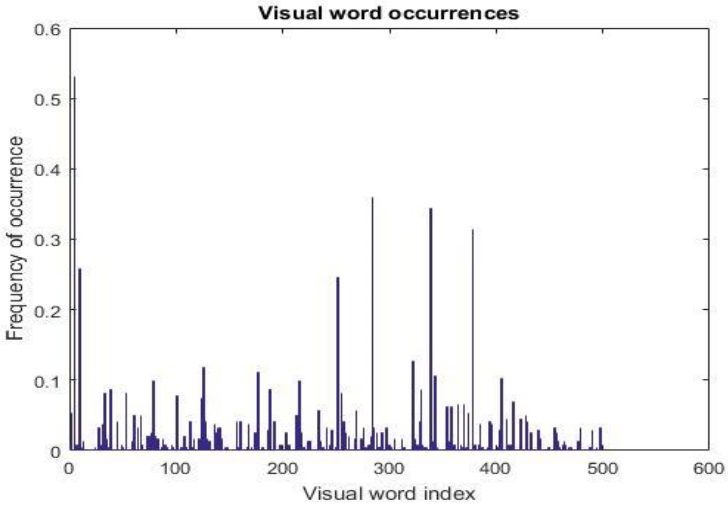


Figure 4.13 - Feature vector from Bag of Visual Words Model for Walking Action

Algorithm 4.6: Proposed Method using MHI and Bag of Visual Words

```
Inputs:
MHI images $M$ – $M$ is $n\ x\ m$ image
$K$ – number of clusters for K-Means

Outputs:
Feature Vector $V$ – $V$ of size 1 $x\ n$

1: Initialize variables:
        $K \rightarrow 500$ (Number of clusters for k-means)
        $MHI \rightarrow\ M$ (M is a single MHI image)
2: Split MHI into training and testing sets
3: For each $MHI$ in each training action set
        $V \rightarrow extract\ feature\ vector\ from\ bag\ of\ visual\ words$ (see section 4.4)
        $label \rightarrow assign\ label\ for\ current\ action$
   End For
4:  Train feature vector $V$ using SVM.
```

4.8 Conclusion

In this chapter, the proposed method was introduced. It combines MHI with BOVW in order to recognize human actions. MHI using BOVW first extracts SURF features from the MHI and then clusters them using K-Means. The resultant features are then learned using SVM. It is clear that by using the SURF approach in the BOVW model, the features extracted from MHI is scale, rotational and view point invariant and is computationally simple. This is more descriptive as compared to using LBP which just gives information about the direction of motion. In the LBP using MHI approach, LBP features are extracted from the MHI, the histogram of the resultant LBP image is then computed and thus resulting in a feature vector which is then used to train SVM and KNN. This chapter also introduced various feature extraction techniques namely; SURF, SIFT, LBP and BOVW used in the human action recognition domain. The BOVW is used to extract key points using the SURF approach from an image that are combined to form the base of a novel approach to the human action recognition problem in this work. The extraction of SURF features from MHI have enabled us to obtain a motion template that has the potential of boosting the recognition rate of human action recognition.

CHAPTER 5 - EXPERIMENTAL RESULTS AND DISCUSSION

5.1 Introduction

The experiment investigates the effects of MHI and its variations using BOVW and the LBP feature extraction methods that form the proposed method. The investigation also aims to show the effects of the relevant MHI parameters and how it affects the overall recognition rate when used in the human action recognition domain. In section 5.2 the experimental setup is described. Section 5.3 describes the testing procedures used to evaluate the proposed approaches. In Section 5.4 experiments are conducted to show the effects that each parameter in the MHI method has on preserving movement history of a silhouette. Section 5.5 shows the results of the MHI using LBP method. Section 5.6 shows the results obtained using MHI and the BOVW approach. Section 5.7 summarizes the results obtained in this work. Section 5.8 compares the results obtained to the state-of-the-art.

5.2 Experimental Setup

In all our experiments, the Weizmann and KTH datasets are used to validate the proposed methods. The machine used is an intel i5 4500U at 2.60 GHz with 4GB RAM. The experiment is split into three sections, which are:

- Effects of MHI with its relevant parameters
- MHI using Local Binary Patterns (LBP)
- MHI using Bag of Visual Words (BOVW)

5.2.1 Effects of MHI with Its relevant Parameters

This experiment aim to show the amount of history information that is present when using each variant of the MHI approach. This is done by changing its relevant parameters. In the original MHI approach, the value of τ is varied. In the Modified MHI method, τ is set to a constant value (150) and the δ parameter is varied. In the Timed MHI approach the parameter α is varied. By showing this, we can then identify what values are required to obtain a good representation of the history of an action motion for a given dataset.

5.2.2 MHI using LBP

This experiment shows the results obtained using MHI with the LBP approach to solve the human action recognition problem. Training is done using K-Nearest Neighbours and Support Vector Machines. The hold-out validation approach is used to validated this approach by employing an 80-20 split for training and testing respectively. Training is then done five times. The average of the results is computed and plotted. Training Parameters are as follows:

- Local Binary Patterns
    - Radius = 1
    - Sampling Points = 8
- K-Nearest Neighbours
    - K = 2
- Support Vector Machine
    - Multiclass SVM
    - Kernel – Linear

The training parameters chosen gave the most optimal system performance. These were chosen based upon previous works from the literature. The aim was to keep these parameters constant throughout the experimental setting in order to test the effects of the various parameters of the MHI algorithm.

5.2.3 MHI using BOVW

This experiment evaluates the proposed method which is combining MHI with the BOVW approach. The hold-out validation approach is used with an 80-20 split used for training and testing respectively. Training is done 5 times in order to randomize both the test and training sets. The average of the 5 trials are then computed and plotted.

SURF features are extracted with a grid step of [8 8] and a Block Width of [32 64 96 128]. In the K-Means clustering step, K is set to 500.

Training is done using SVM with the following parameters:

- Multiclass SVM
- Kernel – linear

5.3 Testing Procedures

5.3.1 Hold Out Validation

The hold-out validation approach was chosen for validating the two proposed methods. The hold-out validation approach splits up the dataset into two sets, these are training and test sets. The training set is what the model is trained on, and the test set is used to see how well the model performs. In the machine learning domain, a common split of 80% for training and 20% for testing is used. In order to show the performance of the proposed method in this dissertation the hold-out validation approach is run 5 times; this randomizes the training and test sets. The average of the recognition rate is then taken as the final recognition rate for evaluating the model. The accuracy for each trial is computed as follows [1]:

$$Accuracy\ (\%) = \frac{Number\ of\ correctly\ identified\ Images\ in\ test\ set}{Total\ Number\ of\ Images\ in\ test\ set} * 100 \tag{5-1}$$

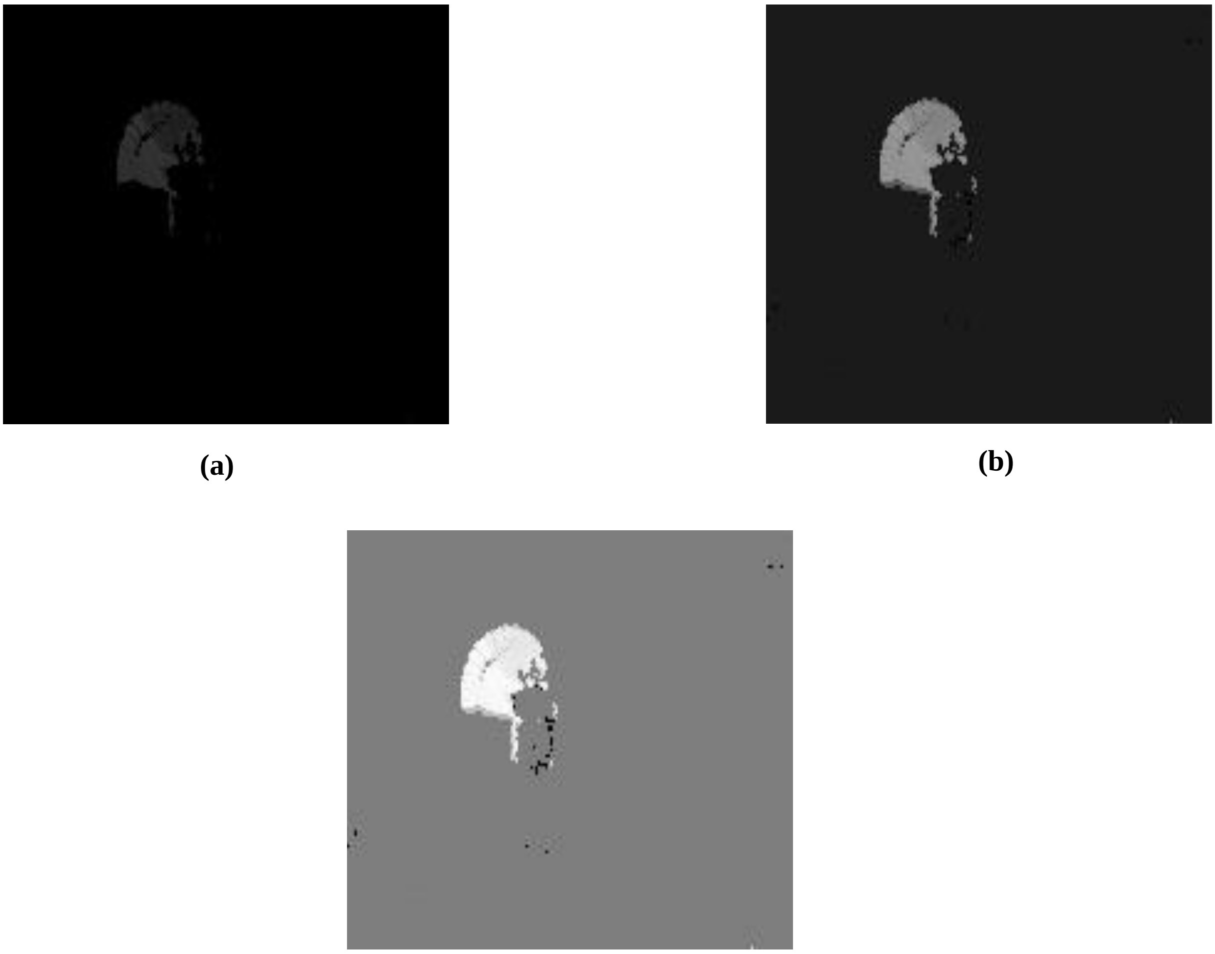
5.4 Motion History Images and Its variations

In this section, the experiment aim to show the effects of the various parameters of MHI and the impact they have on the amount of information stored for the moving regions.

5.4.1 Original Motion History Images

MHI are extracted and shown in Figure 5.1 below. Various values of τ are used to generate the images. The choice of the value of τ influences performance when generating MHI.

It can be seen from Figure 5.1 that increasing the value of τ, increases the intensity of the pixels of the image. The increase in the τ value also stores more movement information. Therefore, considering a value for τ should be based on the number of frames a video has or number of images in a sequence. Having a τ value which is too high or too low may hinder the overall performance of the system. The effects of this on recognition rates can be seen in sections 5.5 and 5.6 of this chapter.



(a)

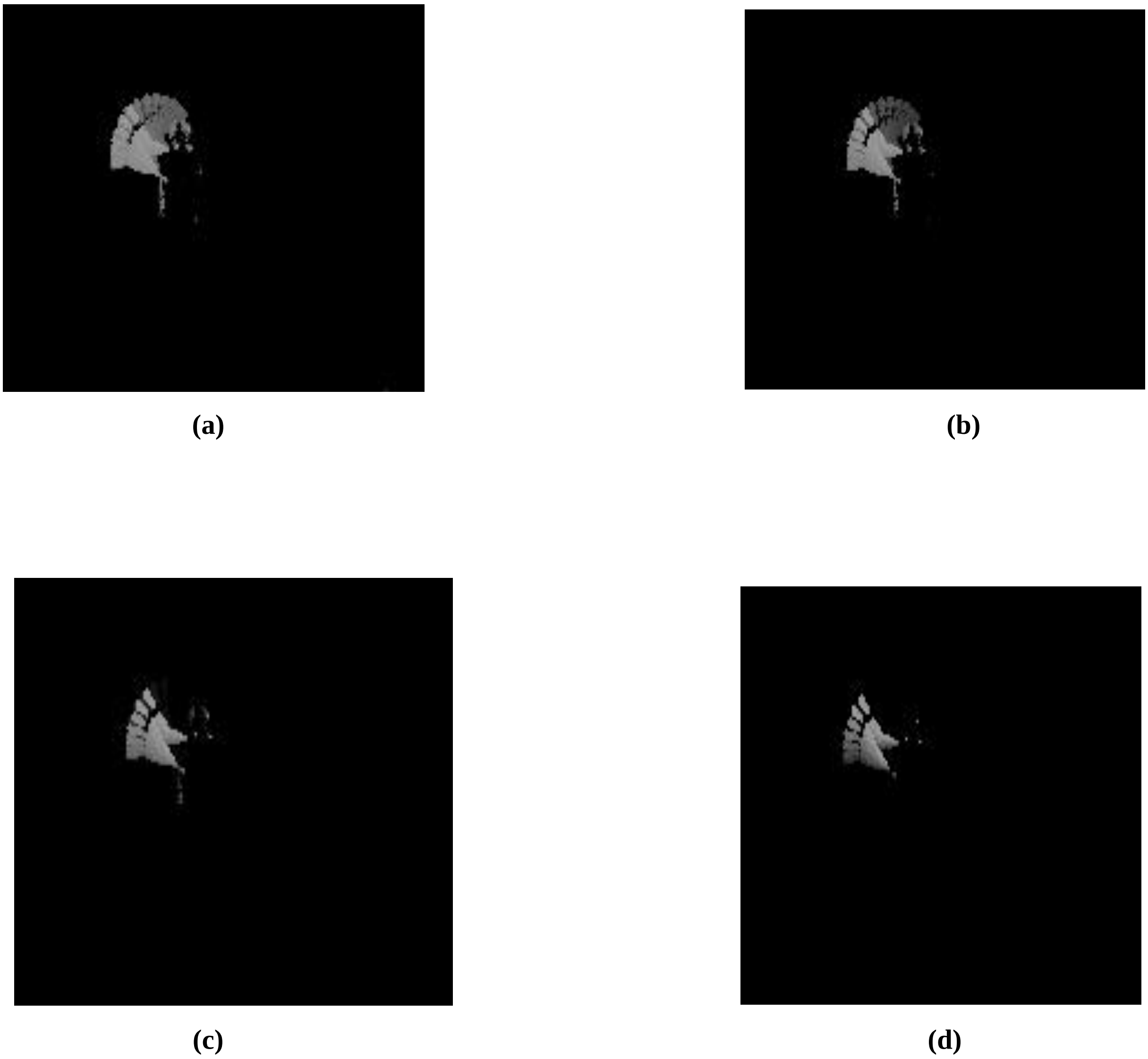
(b)

(c)

Figure 5.1 – One Hand Waving Action: (a) Tau = 50, (b) Tau = 150, (c) Tau = 250

5.4.2 Modified Motion History Images

The Modified Motion History approach employs a δ value in the equation. By increasing the δ value, as seen in Figure 5.2 below, more information is lost from the silhouette. This modified approach of MHI, allows for τ to be constant and minimizes the effort involved in going through all τ values. i.e. 0 – 255 to find the optimum solution. δ values are usually small and have a range between 1 and 10 for higher recognition rates. τ can be set to a constant value and by using the δ parameter, the amount of information needed can be fine-tuned. The effects of δ can be seen in Figure 5.2. The effect on recognition rates can be seen in sections 5.5 and 5.6.



(a)

(b)

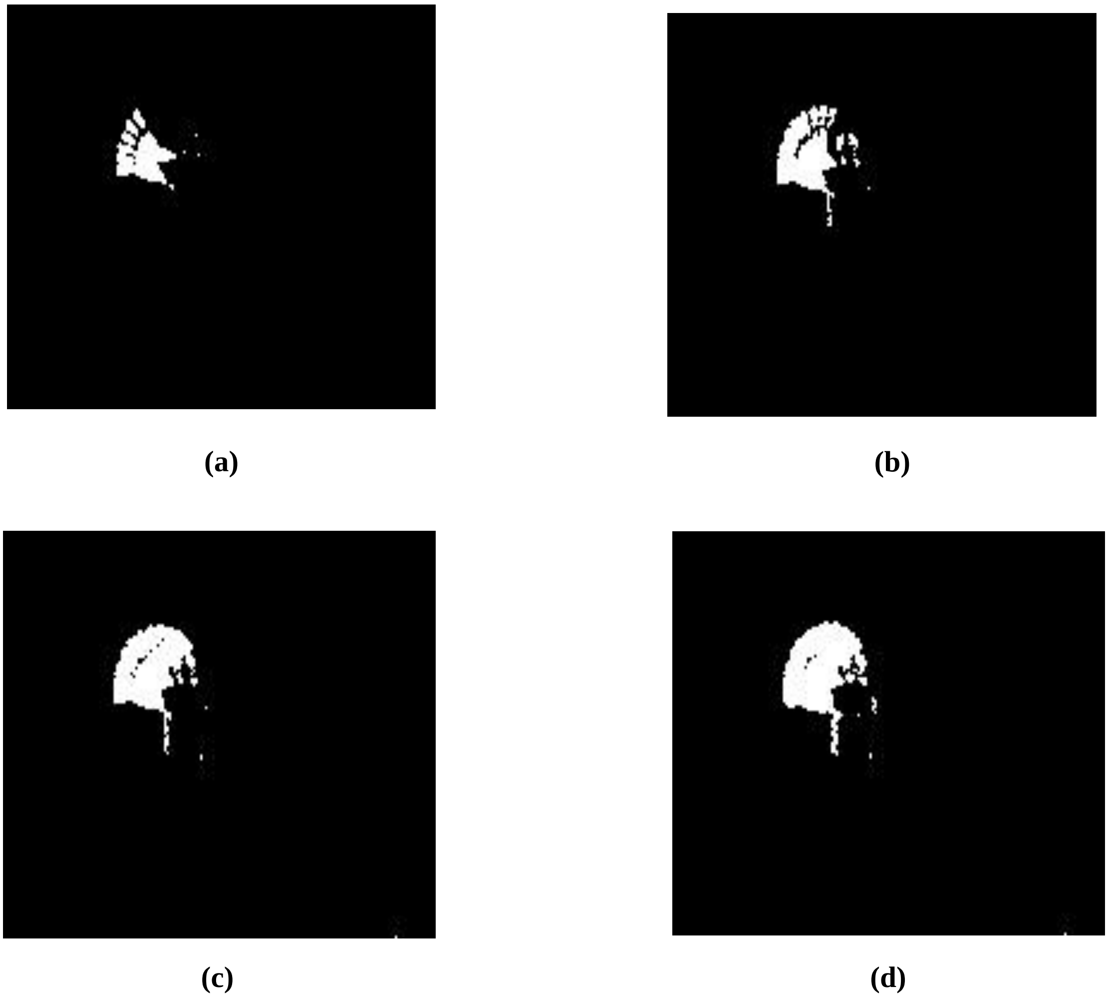
(c)

(d)

Figure 5.2 – One Hand Waving Action: (a) Delta = 3, (b) Delta = 5, (c) Delta = 10, (d) Delta = 20 (Tau = 150 for all images*)

5.4.3 Timed Motion History Images

In the Timed MHI approach, the parameter α symbolises the maximum duration. The effects of changing α can be seen in the Figure 5.3 below. It is clear that by increasing the value of α, more history is preserved. In the human action recognition application, choosing a value of α may be vital to achieving good recognition rates for this variant of the MHI approach.



(a) (b) (c) (d)

Figure 5.3 – One Hand Waving Action: (a) Alpha = 0.5s, (b) Alpha = 1.0s, (c) Alpha = 2.0s, (d) Alpha = 4.0s

5.5 Motion History Images using Local Binary Patterns

In this section, the three variants of MHI mentioned in this work are tested using the Local Binary Patterns approach. Local Binary Patterns are used to extract feature vectors, which are then trained using SVM and KNN. Results for each dataset and MHI variant are obtained and plotted.

5.5.1 Results for Weizmann Dataset

5.5.1.1 Original Motion History Image

From the results in Figure 5.4 below, it can be seen that the two classification algorithms yield similar results. For K-Nearest Neighbours the highest recognition rate achieved is 57% when τ is 150. Using Support Vector Machines, a recognition rate of 60% is achieved when τ is set to 100. The recognition rates shown are rather low, this could be due to the fact that the running, walking, jumping and side motions in the Weizmann dataset yield the same Original MHI and therefore extracting LBP features will yield almost similar histograms for training and testing.

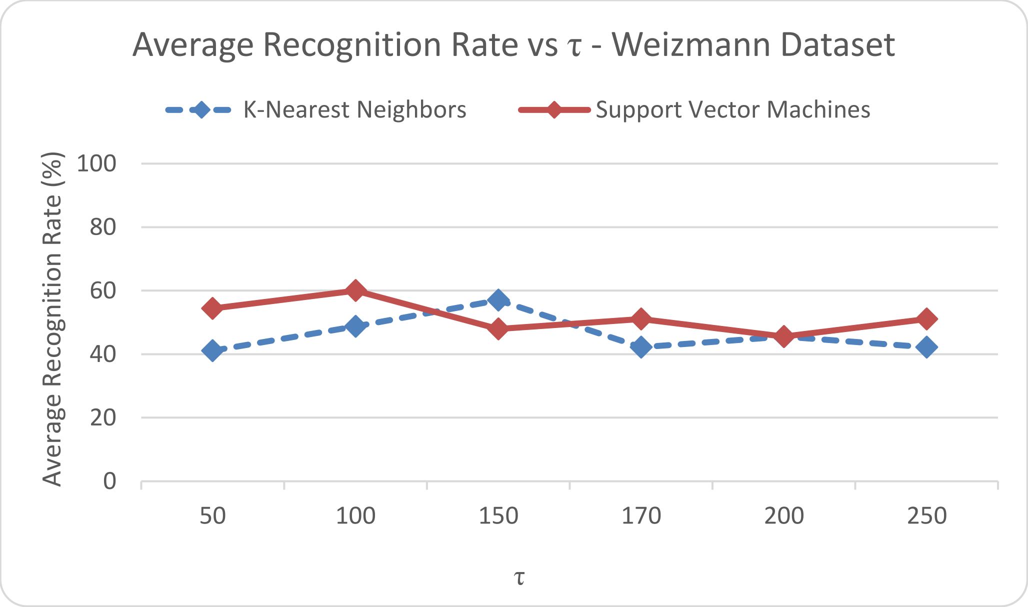


Figure 5.4 - Average Recognition Rate (%) vs τ for Weizmann Dataset - Original MHI using LBP

5.5.1.2 Modified Motion History Image

It can be seen in Figure 5.5 that the two classification methods yield a similar trend in recognition rates for the modified MHI approach. By changing δ values when τ is 150, the recognition rate improves significantly. This shows that when designing an action recognition model based on MHI, τ and δ play an important role in obtaining higher recognition rates. The highest recognition rate is obtained when δ is set to 3 for both classification methods. A recognition rate of 61% is obtained for Support Vector Machines and 48% is achieved for K-Nearest Neighbours. It can also be seen that the recognition rate reaches its peak at δ = 3 and slowly decays as δ is increased, this is so as increasing δ removes more motion information in the Modified MHI approach as discussed in section 5.4.2

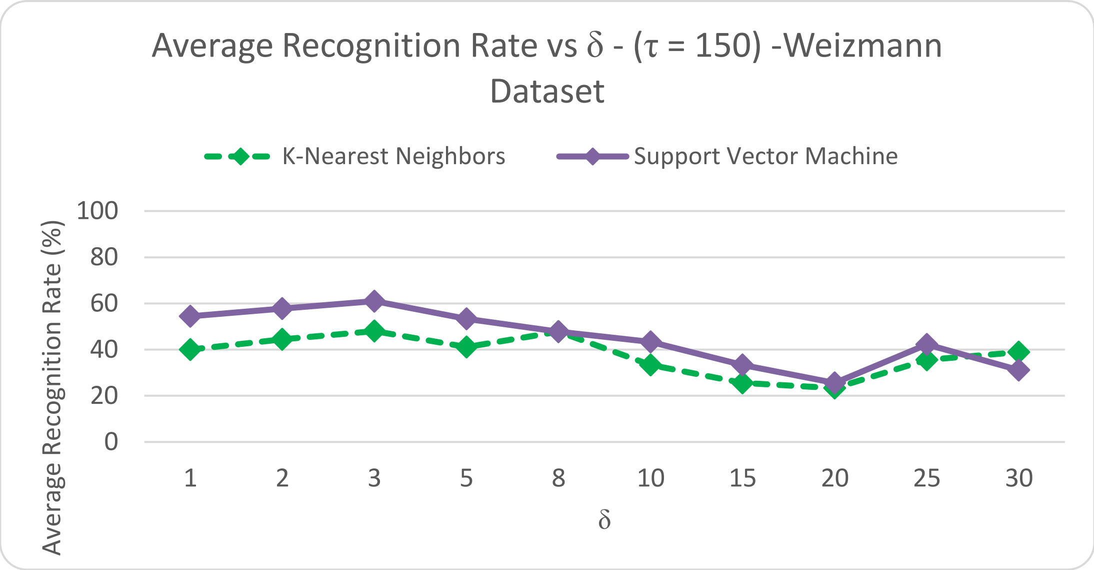


Figure 5.5 - Average Recognition Rate (%) vs δ for Weizmann Dataset - Modified MHI using LBP

5.5.1.3 Timed Motion History Image (TMHI)

Figure 5.6 shows that the TMHI approach also yields poor recognition rates. It can be seen that changing the duration parameter α on the Weizmann dataset yields almost similar recognition rates for both classifiers.

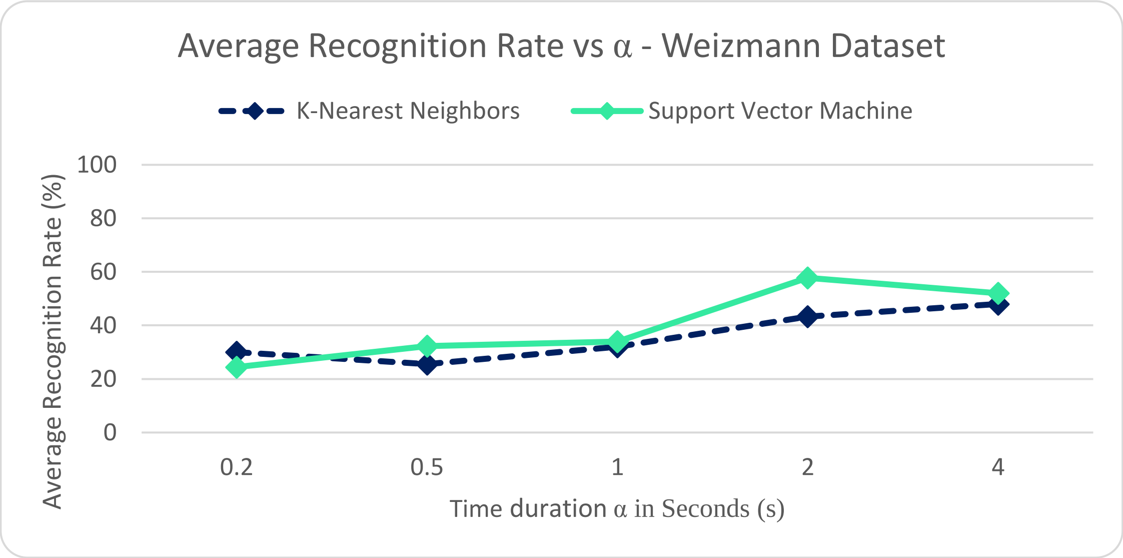


Figure 5.6 - Average Recognition Rate (%) vs α for Weizmann Dataset - timed MHI using LBP

The highest recognition rate for KNN is achieved when α is set to 4.0s with a recognition rate of 48%. The SVM classifier's highest recognition rate is when α is set to 2.0s with a recognition rate of 57.78%. The recognition rate improves by almost 15%, with the SVM classifier when α is 2.0s compared to the KNN classifier.

5.5.2 Results for the Kth Dataset

The trend for low recognition rates for the LBP method is also observed for the KTH dataset.

5.5.2.1 <u>Original Motion History</u>

From Figure 5.7 we can observe that the KNN classifier outperforms the SVM classifier for all values of τ on the KTH dataset. The highest recognition rate is achieved when τ is 100 for KNN with a recognition rate of 52.00%. SVM yields a high recognition rate of 35.67% when τ is 170.

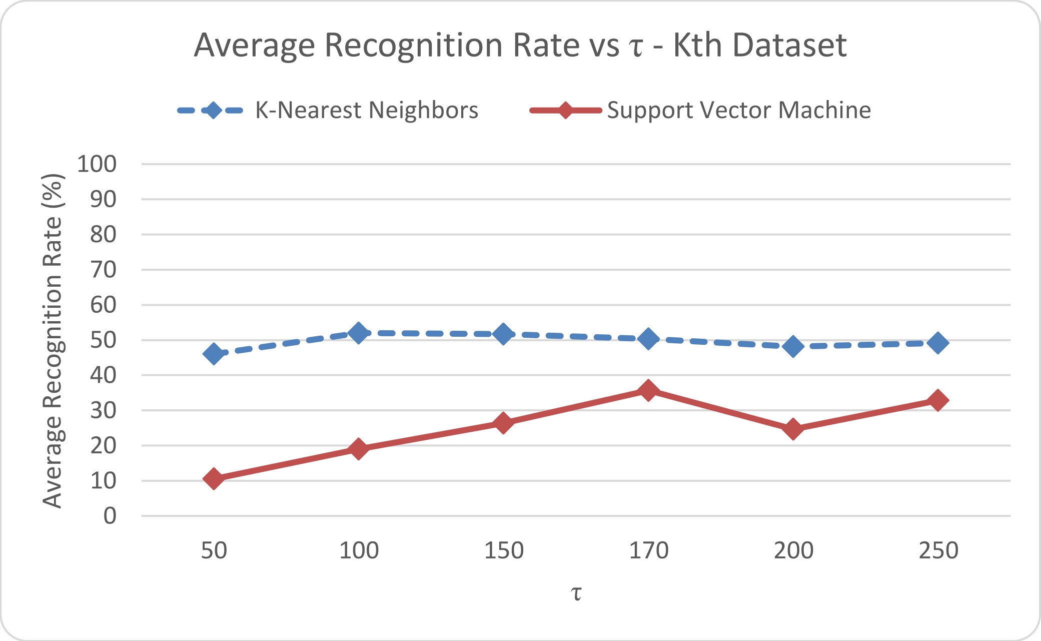


Figure 5.7 - Average Recognition Rate (%) vs τ for KTH Dataset - Original MHI using LBP

5.5.2.2 <u>Modified Motion History Image</u>

In Figure 5.8, it is shown that changing δ values for the KTH dataset does not improve the recognition rate as expected, when using the SVM classifier. By setting δ to 1, the same results are achieved as the Original MHI approach for τ = 150. As δ increase, it is evident that there is downward trend in recognition rates. This is as expected as more pixel's decay resulting in less information being present in the MHI. The highest recognition rates for both classifiers are rather low with SVM producing a recognition rate of 25.166% at δ = 1 and KNN has its highest recognition rate when δ is 3 with a recognition rate of 50.00%.

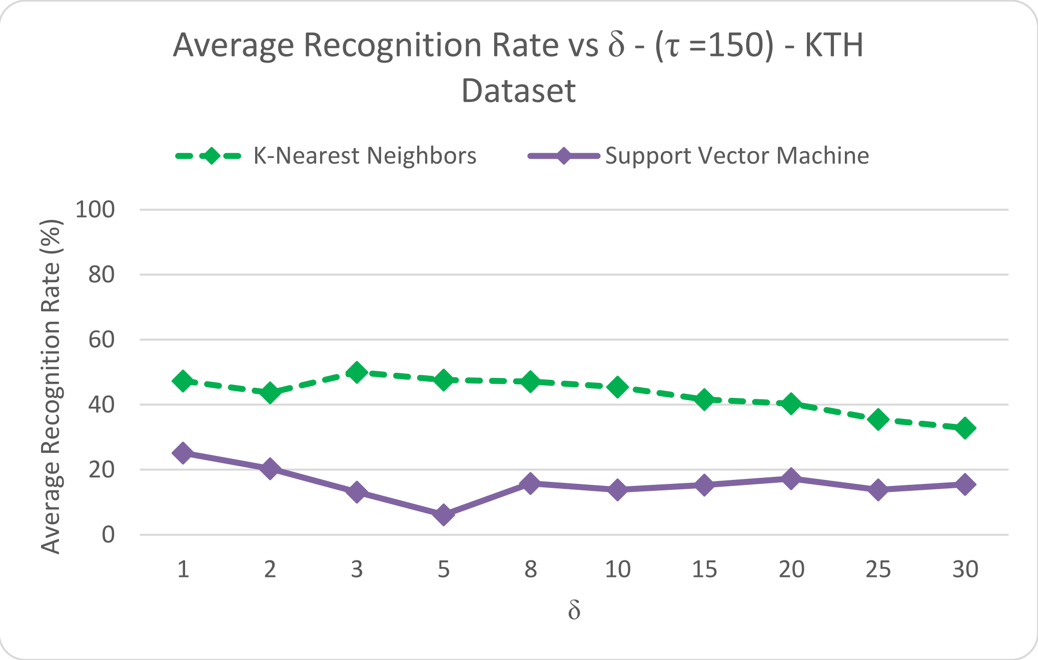


Figure 5.8 - Average Recognition Rate (%) vs δ for KTH Dataset - Modified MHI using LBP

5.5.2.3 Timed Motion History Image

Figure 5.9 shows the recognition rates of the Timed MHI approach using KNN and SVM. It can be seen that the KNN classifier outperforms SVM when using the LBP method. The highest recognition achieved is 49.9% for KNN when α is 0.5s and 22.67% for SVM when α is 4.0s. Although there is a big difference in recognition rates achieved between the two classifiers, the LBP approach does not give good results when combined with the Timed MHI approach. This is due to the fact that the Timed MHI for running, walking and jogging for the KTH dataset are similar and this makes recognition of each action much more difficult.

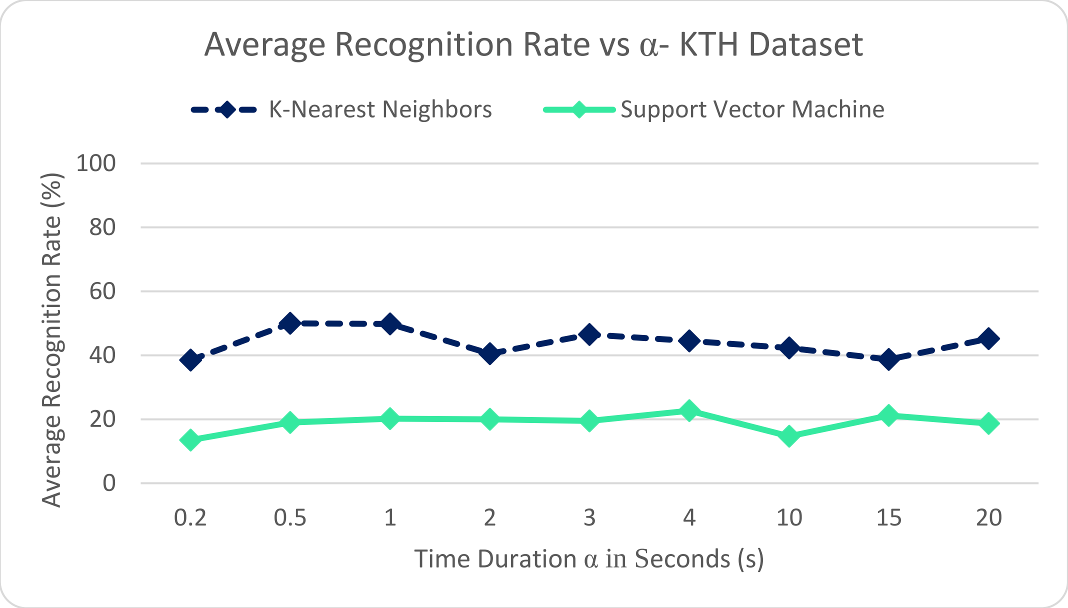


Figure 5.9 - Average Recognition Rate (%) vs α for KTH Dataset - timed MHI using LBP

5.6 Motion History Images using Bag of Visual Words

In this section, the three variants of MHI mentioned in this work are tested using the BOVW approach. BOVW are used to extract feature vectors, which are then learned using SVM.

5.6.1 Results for the Weizmann Dataset

5.6.1.1 Original Motion History Image

Figure 5.10 shows the recognition rate obtained for the Weizmann dataset. The recognition rate is the average of five runs conducted for each value of τ. A value of 50 for τ yields the highest recognition rate of 87%. By changing values of τ, it can be seen that the choice of τ is vital in order to produce an optimal recognition rate. The value of τ should be chosen based on the number of frames in the sequence of images. The choice of the τ value is often neglected in many works however, the results show that this is an important step for human action recognition.

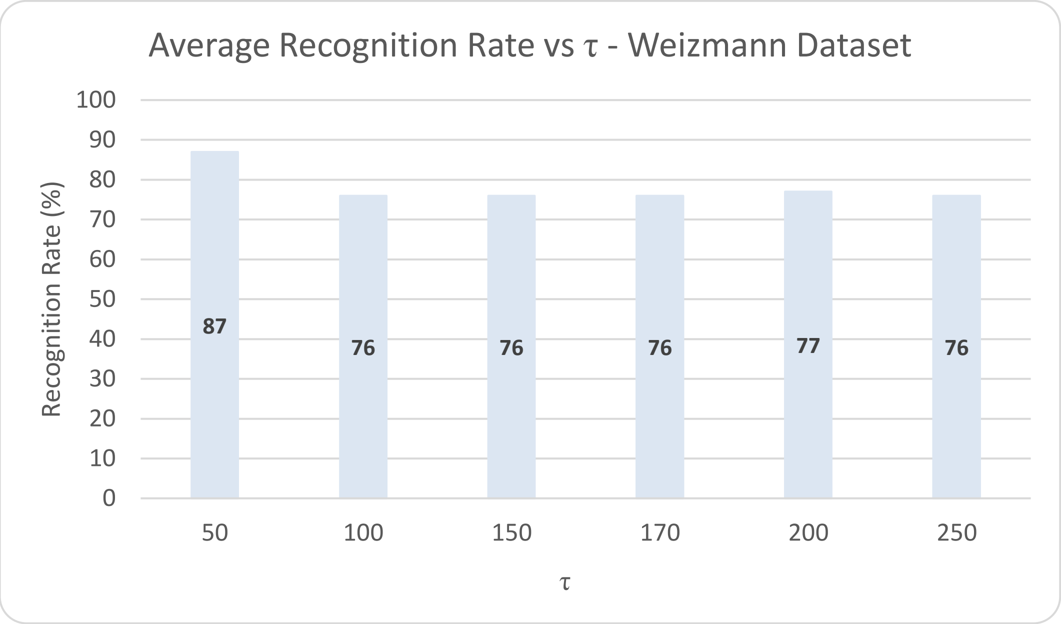


Figure 5.10 - Average Recognition Rate (%) vs τ for Weizmann Dataset – Original MHI using BOVW

5.6.1.2 Modified Motion History Image

In order to test the Modified MHI approach, τ is set to 150. By doing this, the effects of δ can be observed. The results for the different δ values on the Weizmann dataset can be seen in Figure 5.11.

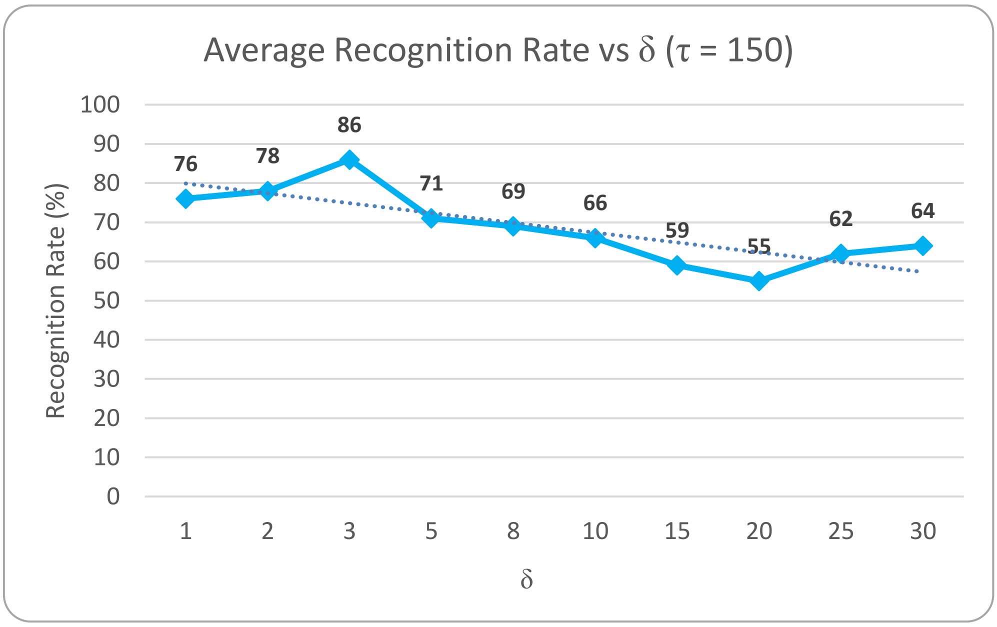


Figure 5.11 - Average Recognition Rate (%) vs δ for Weizmann Dataset - Modified MHI using BOVW

From the original MHI approach, when τ is 150, a recognition rate of 76% was obtained, this is effectively the same as when δ is set to 1, which can be seen in Figure 5.11 above. When δ is set to 3, an improved recognition rate of 86% is achieved. This is a 10% increase in the recognition rate compared to τ = 150 and δ= 1. This shows that by changing the decay parameter δ, the performance of the system can be further optimised.

5.6.1.3 Timed Motion History Image

Since τ represents the current timestamp, there is no need to use a predefined value of τ. This approach to MHI makes the system independent of system speed and frame rates. The duration parameter α is changed and its effects can be seen in Figure 5.12 below.

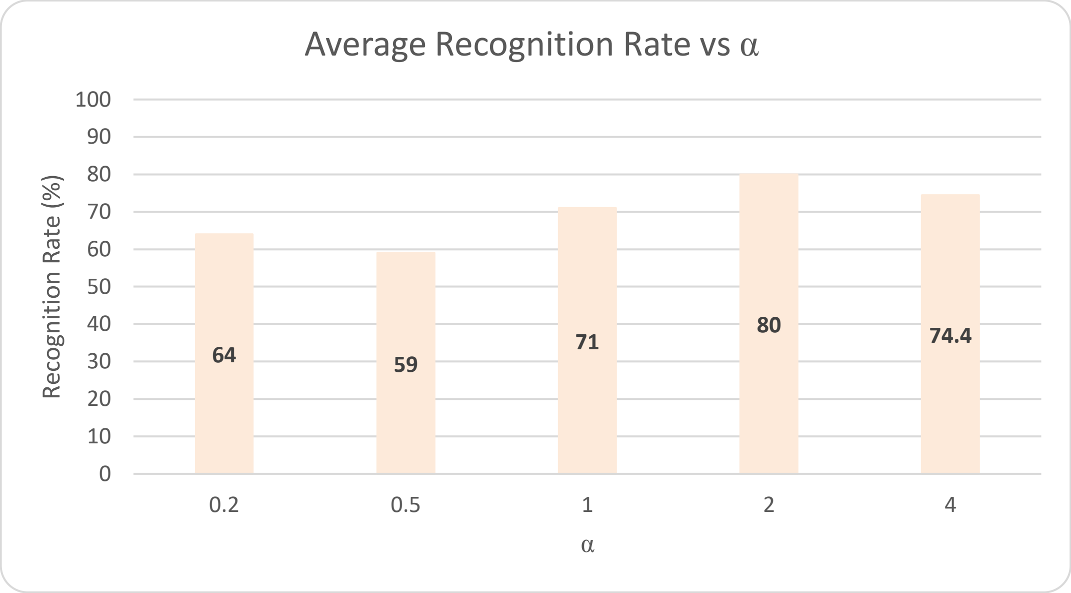


Figure 5.12 - Average Recognition Rate (%) vs α for Weizmann Dataset - timed MHI using BOVW

From the experiments, it is evident that an α value of 2.0s causes the system to have the best recognition rate, which is 80%. Using the TMHI approach, the system can be optimized easily, since α values are not as wide ranging as compared to choosing values of τ and the decay parameter δ in the Original and Modified MHI approaches respectively. The reason for the low recognition rates in this method could be due to the fact that system cannot distinguish between the walk, jump, run and side movements as these movements yield similar MHI.

5.6.2 Results for KTH Dataset

5.6.2.1 Original Motion History Image

The results for the original MHI approach on the KTH dataset can be seen in Figure 5.13 below. At τ = 50, the highest recognition rate of 81% is observed. The results show that there is not much difference in recognition rates when choosing different values of τ for the KTH dataset. The KTH dataset is more challenging as there is a lot of noise in the images generated. Thus, resulting in lower recognition rates.

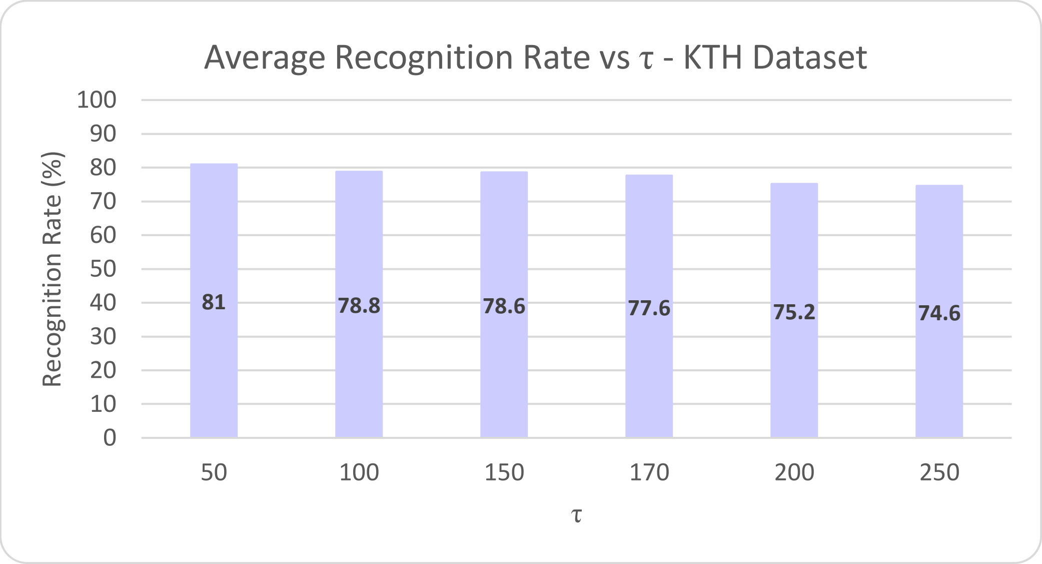


Figure 5.13 - Average Recognition Rate (%) vs τ for KTH Dataset - Original MHI using BOVW

5.6.2.2 Modified Motion History Image

The results for the Modified MHI can be seen in Figure 5.14. It can be seen that the trend for lower recognition rates as δ increases is noted with the KTH dataset, this was also observed with the Weizmann dataset. The best recognition rate is achieved when δ is set at 2 and 3. In the original case for τ = 150, the recognition rate is 78,6%. This is again effectively the same as setting δ to 1. By changing δ to 3, the recognition rate is further improved by 3%.

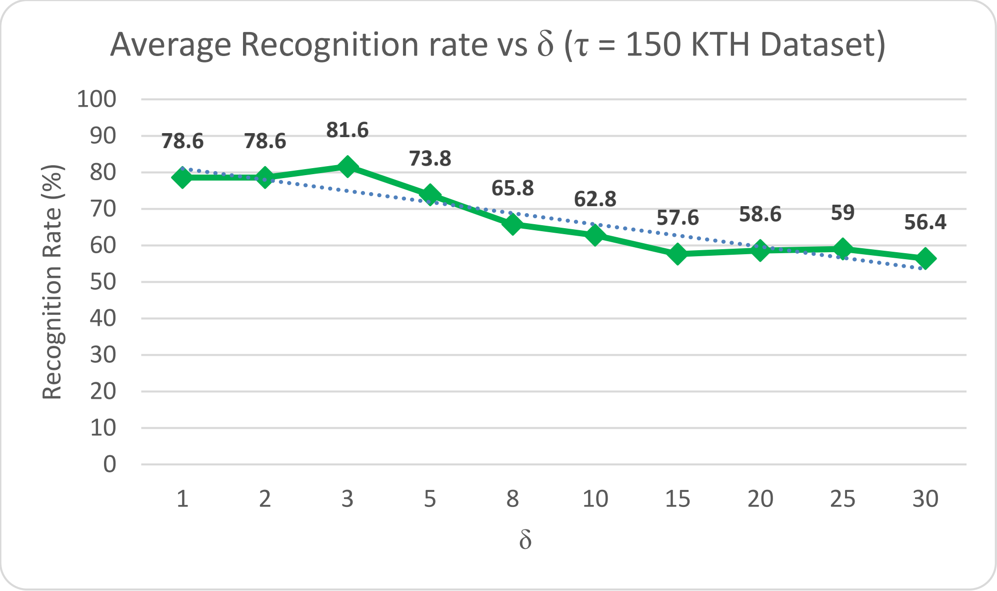


Figure 5.14 - Average Recognition Rate (%) vs δ for KTH Dataset - Modified MHI using BOVW

5.6.2.3 Timed Motion History Image

Figure 5.15 shows that for the KTH dataset, the results are almost similar in terms of recognition rates, when using the Timed MHI approach. The highest recognition rate achieved is 73,8% when α is 0,2s. The low recognition rates could be due to activities such as Jogging, Walking and Running sharing similar MHI. Noise in the KTH dataset could also be a contributing factor for the low recognition rates.

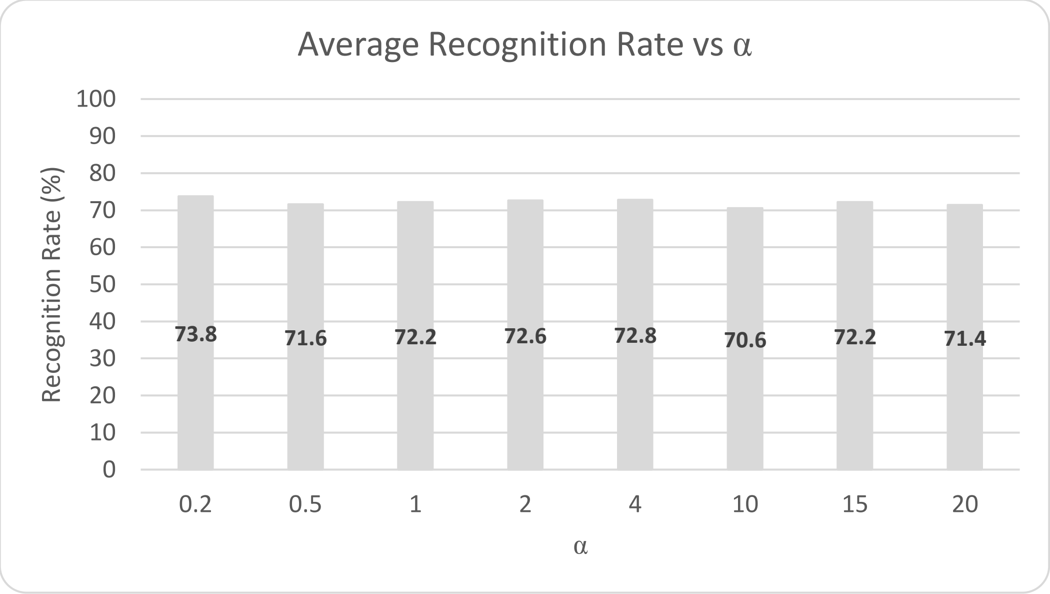


Figure 5.15 - Average Recognition Rate (%) vs α for KTH Dataset - timed MHI using BOVW

5.7 Summary of Results

Table 2 below shows the highest recognition rates achieved for each of the proposed methods. The BOVW model out-performs the LBP approach for all three variants of MHI.

Table 2 - Summary of Results for the Highest Recognition Rates Achieved

Method	Weizmann	Kth
Original MHI using LBP	60%	52%
Modified MHI using LBP	61%	50.02%
Timed MHI using LBP	57.78%	49.99%
Original MHI using BOVW	87%	81%
Modified MHI using BOVW	86%	81.6%
Timed MHI using BOVW	80%	73.8%

The highest recognition rate achieved for the Weizmann dataset is 87% when using the Original MHI method with the BOVW model. This was achieved by using a τ value of 50. The modified MHI approach using the BOVW approach achieves the highest recognition of 81.6% on the KTH dataset by using a τ value of 150 and δ value of 3.

5.8 Comparison with the State of the Art

When compared to the state-of-the-art, the proposed method shows promising results. Table 3 shows a comparison of the proposed method with the current literature for the Weizmann dataset. It can be seen that the highest recognition rate achieved for the proposed method is 87%. This is lower than the method proposed by Kellokumpu [1] of 98.6%. The reason for this is that the proposed method may have difficulty in distinguishing between the walking, running, jumping and side actions from the Weizmann dataset. The proposed method, however performs much better than the method presented by Niebles [41].

Table 3 - Comparison with Results from Literature (Weizmann Dataset)

Source	Accuracy
Niebles and Li [41]	72.8%
Kellokumpu et al. [1]	98.7%
Proposed method with Original MHI (τ = 50) using BOVW	87%

Table 4 - Comparison with Results from Literature (KTH Dataset)

Source	Accuracy
Niebles et al. [42]	83.33%
Kellokumpu et al. [1]	90.8%
Proposed Method with Original MHI (τ = 50) using BOVW	81%
Proposed method with Modified MHI (τ = 150, δ = 3) using BOVW	81.6%

From Table 4, it can be seen that the proposed method achieves a recognition rate of 81.6% which is close to what Niebles et al. [42] achieved. The reason for Kellokumpu et al. [1] achieving a higher recognition rate of 90.8% compared to the proposed of method of 81.6% is that the proposed method does not use a state-of-the-art background subtraction method to eliminate the noise and shadows that the KTH dataset has. Though the frame differencing background subtraction method used in our approach is sufficient enough to compute an MHI, it cannot deal very well with shadows presented in more complex datasets such as KTH. The proposed method also has difficulty distinguishing between the Running, Walking and Jogging actions as these result in similar MHI.

5.9 Conclusion

Experiments have shown that the BOVW approach out-performs the LBP method for human action recognition when SVM is used. The LBP approach only captures information about the direction of motion from the MHI which is not enough to deal with actions that share similar MHI. By using SURF in the BOVW model, the computation is fast and robust to scale, rotation, illumination and is viewpoint invariant. The BOVW features extracted from the MHI performs well when compared to some existing methods from the literature. Both approaches, however, seem to struggle to distinguish between walking, running, and jogging/side movements from both the KTH and Weizmann datasets.

CHAPTER 6 - CONCLUSION

6.1 Summary of Dissertation

There have been vast improvements in the human action recognition domain over the years. However, there is still a lot of work and research required to find a solution that is fool-proof for any activity/action that is being performed. The signs from the current literature are promising. MHI is one of the more famous techniques used for action recognition. It considers the entire silhouette as a whole, and therefore is considered as a holistic approach. MHI keeps record of the history of motion of a silhouette. Subsequent techniques used with the MHI approach look at the entire silhouette of MHI for recognizing human actions.

This research focused on two aspects for action recognition. The first proposes two approaches for action recognition, MHI using LBP and MHI using BOVW. The second aspect was to investigate the effects of the variants of MHI and their relevant parameters play on recognition rates. The variants of MHI presented are: Original MHI, Modified MHI and Timed MHI. The choice of these parameters plays a vital role in the overall recognition rate of a system when using MHI for action recognition. Results from the experiments are promising when compared to the state-of-the-art. When using the MHI method, it is vital to pay attention to its relevant parameters as this could affect the recognition rates of the entire system.

The results of this research show that the BOVW model outperformed the LBP approach. The highest recognition rate achieved for the BOVW is 87% on the Weizmann dataset and for the KTH dataset, a recognition rate of 81.6% is achieved. LBP achieves a high recognition rate of 61% and 52% for the Weizmann and KTH datasets respectively. However, using the BOVW model with MHI may still need to be improved upon. A limitation of the proposed method using BOVW is that it struggles to distinguish between actions that share similar movements such as running, walking and jogging.

6.2 Future Research

Based on the work presented above, we have identified the following future works that can help in improving the results that we have achieved:

- A much more robust background subtraction algorithm such as the Gaussian Mixture Model (GMM) could further improve the recognition rate of the system and quality of the MHI generated.
- The effects of other feature descriptors and detectors, such as SIFT for the BOVW model should be investigated for further improvement of recognition rates.
- Since deep learning is rapidly gaining more interest amongst researchers. Convolutional Neural Networks (CNN) would be well suited for recognizing actions. An interesting experiment would be to feed the MHI extracted from a video into a CNN.
- Using LBP-TOP and discarding the MHI from the LBP method by feeding the unprocessed video stream into LBP-TOP could lead to better recognition rates for the LBP method.

REFERENCES

[1] Matti Pietikäinen, Vili Kellokumpu, Guoying Zhao, “Recognition of human actions using texture descriptors,” *M. Machine Vision and Applications,* vol. 22, no. 5, pp. 768-780, 2007.

[2] Xin Xu, Jinshan Tang, Xiaolong Zhang,Xiaoming Liu, Hong Zhang, Yimin Qiu, “Exploring Techniques for Vision Based Human Activity Recognition: Methods, Systems, and Evaluation,” *Sensors,* vol. 13, pp. 1636-1649, 2013.

[3] M. A. R. Ahad, Motion Histoy Images for Action Recogniton and Understanding, London: Springer, 2013.

[4] P. Turaga, R. Chellappa, V. S. Subrahmanian and O. Udrea, “Machine Recognition of Human Activities: A survey,” *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 18 , no. 11, pp. 1473-1488, Nov 2008.

[5] Klette R., Tee G., “Understanding Human Motion: A Historic Review. In: Rosenhahn B., Klette R., Metaxas D. (eds) Human Motion,” *Computational Imaging and Vision, Springer, Dordrecht,* vol. 36, 2008.

[6] E.-J. Marey, Mouvement, Paris: G. Masson, éditeur, Librairie de l'Académie de Médecine, 1830-1904.

[7] G. Johansson, “Visual perception of biological motion and a model for its analysis,” *Perception & Psychophysics,* vol. 14, no. 2, pp. 201-211, 1973.

[8] G. Johansson, “Visual motion perception,” *Scientific American,* vol. 232, no. 6, pp. 76-88, 1975.

[9] Lasitha Piyathilaka, Sarath Kodagoda, “Human Activity Recognition for Domestic Robots,” *Field and Service Robotics, Springer Tracts in Advanced Robotics,* vol. 105, pp. 395-408, 2015.

[10] Yu Kong, Yun Fu, “Human Action Recognition and Prediction: A Survey,” vol. 13, no. 9, pp. 1806-11230, 9 September 2018.

[11] “Cybicom Atlas Defence,” Cybicom Atlas Defence, 2018. [Online]. Available: http://cadefence.com/helicopter_flight_deck_trainer.html. [Accessed 31 12 2018].

[12] Microsoft, “Microsoft,” Microsoft, 31 12 2018. [Online]. Available: https://developer.microsoft.com/en-us/windows/kinect. [Accessed 31 12 2018].

[13] Z. Zhang, “Microsoft Kinect Sensor and Its Effect,” *IEEE MultiMedia,* vol. 19, no. 2, pp. 4-10, Feb, 2012.

[14] Bobick, A., Davis, J, “The recognition of human movement using Temporal Templates,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 23, no. 3, p. 257–267, 2001.

[15] Bappaditya Mandal, How-Lung Eng, “Regularised Discriminant Analysis for holistic human activity recognition,” *Activity Recognition,* vol. 1, no. 1541-1672, pp. 21-30, 2012.

[16] Byungyun Lee, Sungjun Hong, Heesung Lee, Euntai Kim, “Regularized eigenspace-based gait recogntion system for human identification,” in *6th IEEE Conference on Industrial Electronics and Applications, ICIEA,* Beijing, China, 2011.

[17] Thomas B. Moeslund, Adrian Hilton , Volker Kruger, “A survey of advances in vision-based human motion capture,” *Sensors, Science Direct,* p. 90–126, 2006.

[18] Byungyun Lee, Sungjun Hong, Heesung Lee, Euntai Kim, “Regularized Eigenspace-based Gait Recogntion,” pp. 1966-1970, 2011.

[19] J. Friedman, “Regularized Discriminant Analysis,” *J.Am Statistical Assoc,* vol. 84, no. 405, pp. 165-175, 1989.

[20] Unknown, “Face Recognition with OpenCV — OpenCV 2.4.12.0 documentation,” OpenCV, 06 04 2016. [Online]. Available: http://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html. [Accessed 05 06 2018].

[21] X.D Jiang, B.Mandal and A.Kot, “Eigenfeature Regularization and Extraction in Face Recognition,” *IEEE Trans.Pattern analysis and Machine Intelligence,* vol. 30, no. 3, pp. 383-394, 2008.

[22] Wikipedia, “Wikipedia,” 25 10 2017. [Online]. Available: https://en.wikipedia.org/wiki/Principal_component_analysis. [Accessed 26 10 2017].

[23] Jyotsna E, Akhil P V, Arun Kumar, “Silhouette based human action recognition using PCA and ISOMAP,” *International Journal of Advanced Research in Computer and Communication Engineering,* vol. 2, no. 11, pp. 4192-4198, November 2013.

[24] Shao, Ling and Chen, Xiuli, “Histogram of Body Poses and Spectral Regression Discriminant Analysis for Human Action Categorization,” *British Machine Vision Conference (BMVC), Aberystwyth, UK,* pp. 1-11, 2010.

[25] Liang Wang and David Suter, "Learning and Matching of Dynamic Shape Manifolds for Human Action Recognition," *IEEE Transactions on Image Processing,* vol. 16, no. 6, pp. 1646 - 1661, 6 June 2007.

[26] Gary R. Bradski and James W. Davis, "Motion segmentation and pose recognition with motion history gradients," *Machine Vision and Applications,* vol. 3, no. 13, pp. 174-184, July, 2002.

[27] J. W. Davis, "Hierarchical Motion History Images for Recognizing Human Motion," in *Proceedings IEEE Workshop on Detection and Recognition of Events in Video,* Vancouver, BC, Canada, 2001.

[28] Weinland, D., Ronfard, R and Boyer, E, "Free viewpoint action recognition using motion history volumes," *Computer Vision and Image Understanding,* vol. 104, no. 2-3, p. 249–257, December, 2006.

[29] Sk. Md. Masudul Ahsan, Joo Kooi Tan, Hyoungseop Kim, Seiji Ishikawa, "Histogram of Spatio Temporal Local Binary Patterns for Human Action Recognition," in *SCIS&ISIS 2014,* Kitakyushu, Japan, 2014.

[30] Naiel M, Abdelwahab M and El-Saban M, "Multi-View Human Action Recognition System," *IEEE Workshop on Applications of Computer Vision,* p. 270–275, 2011.

[31] Hongying Meng, Nick Pears, Chris Bailey, "A Human Action Recognition System for Embedded Computer Vision," *Workshop on Embedded Computer Vision,* 2007.

[32] Chuanzhen Li, Yin Liu, Jingling Wang and Hui Wang, "Combining localized oriented rectangles and motion history image for human action," in *Seventh International Symposium on Computational Intelligence and Design*, 2014.

[33] Chin-Pan Huang, Chaur-Heh Hsieh, Kuan-Ting Lai, Wei-Yang Huang, "Human Action Recognition Using Histogram of Oriented Gradient of Motion History," in *International Conference on Instrumentation, Measurement, Computer, Communication and Control*, 2011.

[34] Md. Atiqur Rahman Ahad, T. Ogata, J.K. Tan, H.S. Kim, S. Ishikawa, "Template-based Human Motion Recognition for Complex Activities," in *2008 IEEE International Conference on Systems, Man and Cybernetics (SMC 2008)*, 2008.

[35] D. Li, L. Yu, J. He, B. Sun and F. Ge, “Action recognition based on multiple key motion history images,” *IEEE 13th International Conference on Signal Processing (ICSP), Chengdu,* pp. 993-996, 2016.

[36] Fiza Murtaza, Muhammad Haroon Yousaf, Sergio A. Velastin, “Multi-view Human Action Recognition using Histograms of Oriented Gradients,” in *13th International Conference on Frontiers of Information Technology,* 2015.

[37] Fiza Murtaza, Muhammad Haroon Yousaf, Sergio A. Velastin, “Multi-view human action recognition using 2D,” *The Institution of Engineering Technology,* vol. 10, no. 7, pp. 758 - 767, 2016.

[38] Csurka, Gabriela, Dance Christopher, Fan Lixin , Willamowski Jutta and Bray Cédric, “Visual Categorization with Bags of Keypoints,” *Workshop on Statistical Learning in Computer Vision (ECCV),* pp. 1-22, 2004.

[39] Ivan Laptev , Lindeberg, “Space-time interest points,” in *Proceedings of the Ninth IEEE International Conference on Computer Vision (ICCV’03),* (2003).

[40] C. Schuldt, I. Laptev and B. Caputo, “Recognizing human actions: a local SVM approach,” *Proceedings of the 17th International Conference on Pattern Recognition (ICPR), Cambridge,* vol. 3, p. 32–36, August, 2004.

[41] J. C. Niebles and Li Fei-Fei, “A Hierarchical Model of Shape and Appearance for Human Action Classification,” *IEEE Conference on Computer Vision and Pattern Recognition, Minneapolis, MN,* pp. 1-8, 2007.

[42] Niebles, J.C., Wang, H. and Fei-Fei, “Unsupervised Learning of Human Action Categories Using Spatial-Temporal Words,” *International Journal of Computer Vision,* vol. 79, no. 3, pp. 299-318, March, 2008.

[43] Paul Scovanner, Saad Ali, Mubarak Shah, “A 3-Dimensional SIFT Descriptor and its Application to Action Recognition,” in *Proceedings of the 15th ACM international conference on Multimedia* , Augsburg, Germany , September 2007.

[44] Akila M, Rajeswari R, “Human Action Recognition Using SURF and HOG Features from Video Sequences,” *International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT),* vol. 2, no. 6, pp. 1259-1264, Dec,2017.

[45] Pasquale Foggia, Gennaro Percannella, Alessia Saggese, Mario Vento, "Recognizing Human Actions by a bag of visual words," in *IEEE International Conference on Systems, Man, and Cybernetics* , Manchester, UK , 2013.

[46] Carletti, V. and Foggia, P. and Percannella, G. and Saggese, A. and Vento, M, "Recognition of human actions from RGB-D videos using a reject option," in *International Workshop on Social Behaviour Analysis (SBA-2013),* 2013.

[47] F. Ofli, R. Chaudhry, G. Kurillo, R. Vidal, and R. Bajcsy, "Berkeley MHAD: A comprehensive multimodal human action database," in *IEEE Workshop on Applications on Computer Vision (WACV). IEEE 2013,* 2013.

[48] Parul Shukla K. K. Biswas Prem K. Kalra, "Action Recognition using Temporal Bag-of-Words from Depth Maps," in *IAPR International Conference on Machine Vision Applications,* Kyoto, JAPAN, May 20-23, 2013.

[49] Alper Yilmaz and Mubarak Shah, "Actions As Objects: A Novel Action Representation," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA,* vol. 1, p. 984–989, 2005.

[50] Han Su, Jiayun Zou and Wenjie Wang, "Human Activity Recognition Based On Silhouette Analysis Using Local Binary Patterns," *2013 10th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD),* pp. 924-929, 2013.

[51] M. Singh, A. Basu and M. K. Mandal, "Human Activity Recognition Based on Silhouette Directionality," *IEEE Transactions on Circuits and Systems Video Technology,* vol. 18, no. 9, pp. 280-1292, September, 2008.

[52] Chen Chen, Hong Liu, Jungong Han, "Multi-Temporal Depth Motion Maps - Based Local Binary Patterns for 3-D Human Action Recogntion," *IEEE Access,* vol. 5, pp. 22590 - 22604, 2017.

[53] Duan-Yu Chena, Sheng-Wen Shihb, and Hong-Yuan Mark Liaoa,, "Human Action Recognition Using 2-D Spatio-Temporal Templates," in *ICME 2007 (IEEE),* 2007.

[54] Daniel DeMenthon, David Doermann, "Video Retrieval using SpatioTemporal Descriptors," in *ACM multimedia,* Berkeley, California, USA, 2003.

[55] J. Yamato, J. Ohya, and K. Ishii, "Recognizing human action in time-sequential images using hidden markov model," *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition,* p. 379–385, 1992.

[56] Haitao Wu, Wei Pan, Xingyu Xiong and Suxia Xu, "Human Activity Recognition Based on the Combined SVM&HMM," *Proceeding of the IEEE International Conference on Information and Automation,* pp. 219-224, 2014.

[57] Yangsheng Xu, Jie Yang, and Chiou S. Chen, "Human Action Learning via Hidden Markov Model," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans,* vol. 27, no. 1, pp. 34-44, Jan 1997.

[58] Palwasha Afsar, Paulo Cortez, Henrique Santos, "Automatic Human Action Recognition from Video using Hidden Markov Model," in *IEEE 18th International Conference on Computational Science and Engineering*, 2015.

[59] Z. Qu, T. Lu, X. Liu, Q. Wu and M. Wang, "A new method for human action recognition: Discrete HMM with improved LBG algorithm," in *2015 IEEE 9th International Conference on Anti-counterfeiting, Security, and Identification (ASID),* Xiamen, China, 2015.

[60] Kha, Pham The Hai and Ha Hoang, "An Efficient Star Skeleton Extraction for Human Action Recognition Using Hidden Markov Models," *IEEE Sixth International Conference on Communications and Electronics (ICCE), Ha Long,* pp. 351-356, 2016.

[61] Siqi Nie and Qiang Ji, "Capturing Global and Local Dynamics for Human Action Recognition," *22nd International Conference on Pattern Recognition, Stockholm,* pp. 1946-1951, 2014.

[62] J. Wang, Z. Liu, Y. Wu, and J. Yuan., "Mining actionlet ensemble for," in *IEEE Conference on Computer Vision and Pattern Recognition, CVPR* , Providence, RI, United States, 2012.

[63] Tongwei Lu, Ling Peng, "Human Action Recognition of Hidden Markov Model Based on Depth Information," in *15th International Symposium on Parallel and Distributed Computing,* 2016.

[64] Zhimeng Zhang, Xin Ma, Rui Song, Xuewen Rong, Xincheng Tian, Guohui Tian, Yibin Li, "Deep Learning Based Human Action Recognition:A Survey," in *Chinese Automation Congress (CAC), 2017, pp. 3780-3785. IEEE, 2017.,* 2017.

[65] Max Wang, Ting-Chun Yeh, "Human Action Recognition Using CNN and BoW Methods," Stanford University, Stanford, CA, 2016.

[66] Chao Li, Shouqian Sun, Xin Min, Wenqian Lin, Binling Nie, Xianfu Zhang, “End-to-End Learning of Deep Convolutional Neural Network for 3D Human Action Recognition,” in *Proceedings of the IEEE International Conference on Multimedia and Expo Workshops (ICMEW)*, 2017.

[67] Shuiwang Ji, Wei Xu, Ming Yang, Kai Yu, “3D Convolutional Neural Networks for Human Action Recognition,” in *27th International Conference on Machine Learning*, Haifa, Isreal, 2010.

[68] Chuankun Uh, Pichao Wang, Shuang Wangll J, Yonghong Houl, Wanqing U, “Skeleton-Based Action Recognition using LSTM and CNN,” in *Proceedings of the IEEE International Conference on Multimedia and Expo Workshops (ICMEW) 2017*, 2017.

[69] Binu M Nair and Dr. Vijayan K Asari, “Regression based Learning of Human Actions from Video using HOF-LBP Flow patterns,” in *IEEE International Conference on Systems, Man, and Cybernetics*, 2013.

[70] Pavel Zhdanov , Adil Khan, Adín Ramírez Rivera , Asad Masood Khattak, “Improving Human Action Recognition through Hierarchical Neural Network Classifiers,” in *2018 International Joint Conference on Neural Networks (IJCNN)* , Rio de Janeiro, Brazil, 2018.

[71] Gul Varol, Ivan Laptev, and Cordelia Schmid, Fellow, IEEE, “Long-term Temporal Convolutions for Action Recognition,” *IEEE Transactions on Pattern Analysis and Machine Intelligence* , vol. 40, no. 6, pp. 1510-1517, 06 June 2018.

[72] Chuan Sun, Imran Junejo, Hassan Foroosh, “Action Recognition using Rank-1 Approximation of Joint Self-Similarity Volume,” *International Conference on Computer Vision, Barcelona,* pp. 1007-1012, 2011.

[73] Samsu Sempena, Dr. Nur Ulfa Maulidevi, S.T, M.Sc , Peb Ruswono Aryan,M.T, “Human Action Recognition Using Dynamic Time Warping,” in *International Conference on Electrical Engineering and Informatics,* Bandung, Indonesia, July 2011.

[74] Wing W.Y. NG, Jinde Li, Jianjun Zhang, Qiuxia Wu, Jiayong Li, “Visual Words Selection for Human Action Recognition using RBFNN via the Minimization of L-GEM,” in *International Conference on Wavelet Analysis and Pattern Recognition,* Ningbo, China, 9-12 July 2017.

[75] Song Wang, Jianwu Dang, Yangping Wang, Lixia Liu,Zhenhai Zhang, "Recognizing Human Actions Using 3D Motion Trail Model," in *International Conference of Intelligent Robotic and Control Engineering (IRCE) pp. 252-256,* Lanzhou, Gansu Province, China, 2018.

[76] B.P.L. Lo and S.A. Velastin, "Automatic Congestion Detection System for underground platforms," *Proceedings of 2001 International Symposium on Intelligent Multimedia, Video and Speech Processing. ISIMP 2001, Hong Kong, China,* pp. 159-161, 2001.

[77] Rita Cucchiara, Costantino Grana, Massimo Piccardi , Andrea Prati, "Detecting Moving Objects, Ghosts and Shadows in Video Streams," *IEEE Trans Pattern Anal Mach Intell,* no. 25(10):, p. 1337–1342, 2003.

[78] C. Stauffer, W.E.L. Grimson,, R. Romano, L. Lee, "Using adaptive tracking to classify and monitor activities in a site," in *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No.98CB36231)* , Santa Barbara, CA, USA, 1999.

[79] M. A. R. Ahad, T. Ogata, J. K. Tan, H. S. Kim and S. Ishikawa, "Motion recognition approach to solve overwriting in complex actions," in *8th IEEE International Conference on Automatic Face & Gesture Recognition, Amsterdam, 2008, pp. 1-6.*, Amsterdam, Netherlands, 2008.

[80] M. A. R. Ahad, Computer Vision and Action Recognition: A Guide for Image Processing and Computer Vision Community for Action Understanding, square des Bouleaux, France: Atlantis Ambient and Pervasive Intelligence, 2011.

[81] Alexandra Branzan Albu, Trevor Beugeling, "A Three-Dimensional Spatiotemporal Template," *Journal of Multimedia,* vol. 2, no. 4, pp. 45-54, August, 2007.

[82] Timo Ojala, Matti Pietikäinen, David Harwood, "A comparative study of texture measures with classification based on featured distributions," *Pattern Recognition,* vol. 29, no. 1, pp. 51-59, 1996.

[83] Wikipedia, "Wikipedia," Wikipedia, 7 11 2017. [Online]. Available: https://en.wikipedia.org/wiki/Local_binary_patterns. [Accessed 13 11 2017].

[84] M. Pietikäinen, "Scholarpedia Local Binary Patterns," Scholarpedia, 03 03 2010. [Online]. Available: http://www.scholarpedia.org/article/Local_Binary_Patterns. [Accessed 12 11 2017].

[85] Matti Pietikäinen. Guoying Zhao,Abdenour Hadid, Timo Ahonen, Computer Vision Using Local Binary Patterns, London: Springer , 2011.

[86] Di Huang, Caifeng Shan, Mohsen Ardebilian, Yunhong Wang, and Liming Chen , "Local Binary Patterns and Its Application to Facial Image Analysis: A Survey," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews),* vol. 41, no. 6, pp. 765 - 781, Nov 2011.

[87] Timo Ojala, Matti Pietikäinen and Topi Mäenpää, "Multiresolution Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns," *Transactions on Pattern Analysis and Machine Intelligence IEEE,* vol. 24, no. 7, pp. 971-987, July 2002.

[88] D. G. Lowe, "Object recognition from local scale-invariant features," *Proceedings of the Seventh IEEE International Conference on Computer Vision, Kerkyra, Greece,* vol. 2, pp. 1150-1157, 1999.

[89] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *International Journal of Computer Vision* , vol. 60, no. 2, pp. 91-110, (2004).

[90] Hassaan Ali Qazi,Umar Jahangir, Bilal M Yousuf, Aqib Noor , "Human Action Recognition Using SIFT and HOG," in *2017,* Karachi, Pakistan, International Conference on Information and Communication Technologies (ICICT) .

[91] Al-Badarneh, Jafar & Al-Hawary, Talal & morghem, Abdulmalik & Ali, Mostafa & Al-Gharaibeh, Rami, "Keypoints Extraction for Markerless Tracking in Augmented Reality Applications: A Case Study in Dar As-Saraya Museum," in *ICIEI 2014: International Conference on Information and Education Innovations*, Instanbul, Turkey, 2014.

[92] Herbert Bay, Andreas Ess, Tinne Tuytelaars, Luc Van Gool,, "SURF: Speeded Up Robust Features," *Computer Vision and Image Understanding (CVIU),* vol. 110, no. 3, pp. 346-359, 2008.

[93] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *In Proceedings of the fifth Berkeley symposium on mathematical statistics and probability,* vol. 1, no. 14, pp. 281-291, 1967.

[94] Wikibooks, "Data Mining Algorithms In R/Clustering/K-Means," Wikibooks, 6 9 2017. [Online]. Available: https://en.wikibooks.org/wiki/Data_Mining_Algorithms_In_R/Clustering/K-Means. [Accessed 29 12 2018].

[95] Bappaditya Mandal, Xudong Jiang, alex Kot, “Complete discriminant evaluation and feature extraction in kernal space for face recognition,” *Machine vision and applications,* pp. 35-46, 2009.

[96] M. Blank, L. Gorelick, E. Shechtman, M. Irani and R. Basri, “Actions as Space-Time Shapes,” *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, Beijing,* vol. 2, pp. 1395-1402, 2005.

[97] Belhumeur, Peter & Hespanha, Joao & Kriegman, David, “Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection,” *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 19, no. 7, pp. 711-720, July, 1997.

[98] M. B. Abidine and B. Fergani, “Evaluating a new classification method using PCA to human activity recognition,” *International Conference on Computer Medical Applications (ICCMA), Sousse,* pp. 1-4, 2013.

[99] F. I. Bashir, A. A. Khokhar and D. Schonfeld, “Real-Time Motion Trajectory-Based Indexing and Retrieval of Video Sequences,” *IEEE Transactions on Multimedia,* vol. 9, no. 1, pp. 58-65, Jan, 2007.